Metaphor Detection with Context Enhancement and Curriculum Learning

Kaidi Jia and **Rongsheng Li***
College of Computer Science and Technology, Harbin Engineering University, Harbin 150001, China
dasheng@hrbeu.edu.cn

Abstract

Metaphor detection is a challenging task for natural language processing (NLP) systems. Previous works failed to sufficiently utilize the internal and external semantic relationships between target words and their context. Furthermore, they have faced challenges in tackling the problem of data sparseness due to the very limited available training data. To address these two challenges, we propose a novel model called MiceCL. By leveraging the difference between the literal meaning of the target word and the meaning of the sentence as the sentence external difference, MiceCL can better handle the semantic relationships. Additionally, we propose a curriculum learning framework for automatically assessing difficulty of the sentence with a pre-trained model. By starting from easy examples and gradually progressing to more difficult ones, we can ensure that the model will not deal with complex data when its ability is weak so that to avoid wasting limited data. Experimental results demonstrate that MiceCL achieves competitive performance across multiple datasets, with a significantly improved convergence speed compared to other models. Our model is available at https://github.com/Evilxya/MiceCL.git.

1 Introduction

Metaphor has long been a common phenomenon in language and cognition research(Lakoff and Johnson, 2008). It not only exists in our daily communication, but also plays a key role in the effective understanding of abstract concepts. In our daily life, we often use metaphors to convey emotions, ideas and opinions to make communication more vivid, rich and profound. For example, when we say "time is money", we don't mean it literally as gold, but rather as a metaphor for how precious and finite it is.

*corresponding author

However, metaphor detection is a complex and challenging task. It involves the integration of various contextual cues, considering factors such as vocabulary, grammar and context. This undoubtedly places high demands on the capabilities of models. Metaphor detection is similar to word sense disambiguation, which involves determining whether a target word is used metaphorically or literally. However, unlike word sense disambiguation, which has abundant corpus resources, metaphor detection tasks suffer from very limited data availability, resulting in a severe data sparsity problem.

In summary, the current metaphor detection task faces two major challenges: (1) How to effectively use sentence information such as context, grammar and context; (2) How to solve the problem of data sparsity. These two challenges are not independent of each other. Failing to effectively use sentence information will make the data sparsity problem more severe, and the data sparsity problem will also make it difficult for the model to learn truly effective information. Recent methods(Gong et al., 2020; Choi et al., 2021) partially address the first challenge by encoding sentence information through pre-trained Transformer models and leveraging linguistic rules to enhance their representation of complex contextual meaning and various semantic information. However, they focus too much on designing complex structures to better encode sentence information, but ignore how to make full use of the existing data set in the case of sparse data, so they do not address the second challenge. Also, The model they designed does not make good use of linguistic rules. CLCL(Zhou et al., 2023) attempts to address the second challenge by introducing curriculum learning to make better use of the data by transitioning the model from simple to complex examples. However, CLCL uses manual evaluation of difficulty to measure the difficulty of examples. Not only is it too dependent on expert

Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 2726–2737

knowledge, but what is considered simple by humans may not be considered simple by machine models, that is, difficulty evaluation methods may not be suitable for machine models. Therefore, the second challenge is not well addressed by CLCL.

To address the above two challenges simultaneously, we propose a novel Metaphor detection model named **M**etaphor **I**dentification with **C**ontext **E**nhancement and **C**urriculum **L**earning (MiceCL). It adopts new SPV (Selectional Preference Violation)(Wilks, 1978) and MIP (Metaphor Identification Procedure)(Group, 2007) to better solve the first challenge, which we call IE-SPV (Internal and External Selectional Preference Violation) and M-MIP (Multiple Metaphor Identification Procedure). IE-SPV can make full use of the internal and external semantic relationships of the sentence, so as to better determine whether the target word is reasonable in the context. M-MIP learns the basic and complex differences between the contextual meaning and the literal meaning of the target word through two parts respectively. By combining IE-SPV and M-MIP, we can make full use of complex contextual meaning and various semantic information to solve the first challenge. To address the second challenge, we introduce a curriculum learning framework that automatically computes the difficulty. By using the training loss of the pre-trained model as the difficulty, we eliminate the disadvantage of manually evaluating the difficulty and are able to more accurately reflect the difficulty of the sentence as considered by the machine model. We largely addressed the second challenge by transitioning the model from simple to complex examples to make the best use of the limited data.

Our main contributions are:

- We propose novel semantic representation modules M-MIP and IE-SPV for metaphor detection. The combination of M-MIP and IE-SPV can make full use of the internal and external semantic relationships of the sentence, and effectively capture the basic and complex differences between the target word and the context information.
- We propose a curriculum learning framework to automatically measure the difficulty. By using the training loss of the pre-trained model as the difficulty, we eliminate the disadvantage of manually evaluating the difficulty and largely solve the problem of data sparsity.
- Experiments show that our model outperforms all the previous models and has a large performance improvement compared with the original results. Meanwhile, we provide detailed ablation experiments and analysis to illustrate the effectiveness of our model.
- Our method can greatly improve the convergence speed of the model. While the effect is significantly improved, it can also improve the computational efficiency and reduce the dependence on computing power.

2 Related Work

2.1 Metaphor Detection

Since metaphors are ubiquitous in our daily lives, it is crucial to correctly identify the use of metaphors(Lakoff and Johnson, 2008). Early work used feature-based approaches to identify metaphors(Turney et al., 2011; Broadwell et al., 2013; Tsvetkov et al., 2014; Bulat et al., 2017). Since these methods rely too heavily on manually annotated data, they are unable to handle rare uses of metaphors. To solve this problem, they tried to use other linguistic features such as sparse distributional features(Shutova et al., 2010; Shutova and Sun, 2013) and dense word embeddings(Shutova et al., 2016; Rei et al., 2017). Other studies have adopted RNN-based models like bidirectional LSTM (BiLSTM)(Graves and Schmidhuber, 2005) as the encoder(Gao et al., 2018; Wu et al., 2018), using Word2Vec(Mikolov et al., 2013), Glove(Pennington et al., 2014) and ELMo(Peters et al., 2018) as text input representations. Due to the limitations of shallow neural networks in expressing information, these methods also struggle to deal with complex contextual meaning. Recent approaches are based on Transformers, most of which use pre-trained models such as BERT(Devlin et al., 2019) or RoBERTa(Liu et al., 2019) as base models(Gong et al., 2020; Su et al., 2020). They focus too much on designing complex structures to better encode sentence information, but ignore how to make full use of the existing data set in the case of data sparsity, so they fail to solve the problem of data sparsity.

2.2 Curriculum Learning

Curriculum learning was first proposed by Bengio et al. (2009). Its main idea is to imitate the characteristics of human learning and learn samples from simple to difficult, so that the model can easily

find a better local optimal solution and accelerate the training speed. Therefore, curriculum learning can better deal with limited data and can solve the data sparsity problem to some extent. As the research progresses, curriculum learning is gradually applied to various tasks. In the field of computer vision, curriculum learning has played an important role in image classification(Weinshall et al., 2018), question answering(Li et al., 2020), etc. In the field of natural language processing, curriculum learning is mainly applied to machine translation(Platanios et al., 2019; Liu et al., 2020; Zhou et al., 2021; Zhang et al., 2021).

CLCL(Zhou et al., 2023) introduced curriculum learning into the task of metaphor detection to try to solve the problem of data sparsity, but used the method of manual evaluation of difficulty, that is, manually designing the difficulty measurers. This method relies too much on expert knowledge and is not necessarily applicable to the machine models. Therefore, although curriculum learning can make the most of the limited data, CLCL(Zhou et al., 2023) has certain limitations and does not solve the problem of data sparsity well.

3 MiceCL

In this section, we will introduce our proposed model, which is mainly composed of two parts: (1) IE-SPV and M-MIP modules; (2) Curriculum Learning modules. We use the first to make full use of the semantic relationships and the second to re-arrange the training examples. Figure 1 presents the overall structure of the model and details the structure of the first part, while Figure 2 details the structure of the Curriculum Learning modules at the bottom of Figure 1.

3.1 IE-SPV & M-MIP

3.1.1 Transformer Encoder

Given a sentence S containing the target word w_t and the literal usage S' of the target word, we first use the Transformer encoder to encode the two sentences into sentence vectors, and then further analyze the sentence vectors.

$$h_{L1}, \ldots, h_{Ln} = Enc([CLS], S, [SEP]), \quad (1)$$

$$h_{R1}, \ldots, h_{Rm} = Enc([CLS], w_t, [SEP], S', [SEP]) \quad (2)$$

Among them, [CLS] and [SEP] are two special marker symbols in BERT, which play the role of demarcation. The left part is the vector of the given sentence, from which we extract the sentence meaning vector h_s and the contextual meaning vector h_t of the target word. h_s and h_t is calculated as follows.

$$h_s = \frac{1}{n}\sum_{i=1}^{n} h_{Li} \quad (3)$$

$$h_t = \frac{1}{k}\sum_{i=b}^{k} h_{Li} \quad (4)$$

Where b is the starting position of the target word, and k is the number of tokens that the target word is divided into.

The right part is the literal usage vector of the target word, from which we extract the literal meaning vector h_l of the target word. Unlike h_t, h_l does not need to know the specific position of the target word, because the attention mechanism of the Transformer Encoder will automatically converge the target word to the relevant part(Vaswani et al., 2017). It should be noted that h_s, h_l and h_t all contain [CLS] and [SEP] tokens to ensure the validity of the contrast

3.1.2 IE-SPV

The basic idea of SPV is to identify the metaphoricity of a target word by detecting whether the target word is reasonable in the context. Current mainstream methods believe that they can detect semantic differences between the target word and the context by detecting the difference between the contextual meaning vector of the target word and the sentence vector(Choi et al., 2021; Zhang and Liu, 2022) to reflect whether the target word is reasonable in the context. However, the target word vector h_t is directly encoded from the sentence, which represents the contextual meaning of the target word rather than the literal meaning, so the difference between h_t and h_s cannot fully represent the semantic difference between the target word and the context.

Therefore, we propose a new SPV module based on SPV language rules, which we call IE-SPV (Internal and External Selectional Preference Violation). We take the difference between the contextual meaning vector h_t of the target word and the sentence vector h_s as the sentence internal difference h_{in}, which represents the contextual meaning of the target word and the semantic difference between the words in the sentence. The difference between the sentence vector h_s and the literal meaning vector h_l of the target word is taken as the sentence external difference h_{ex}, which represents the

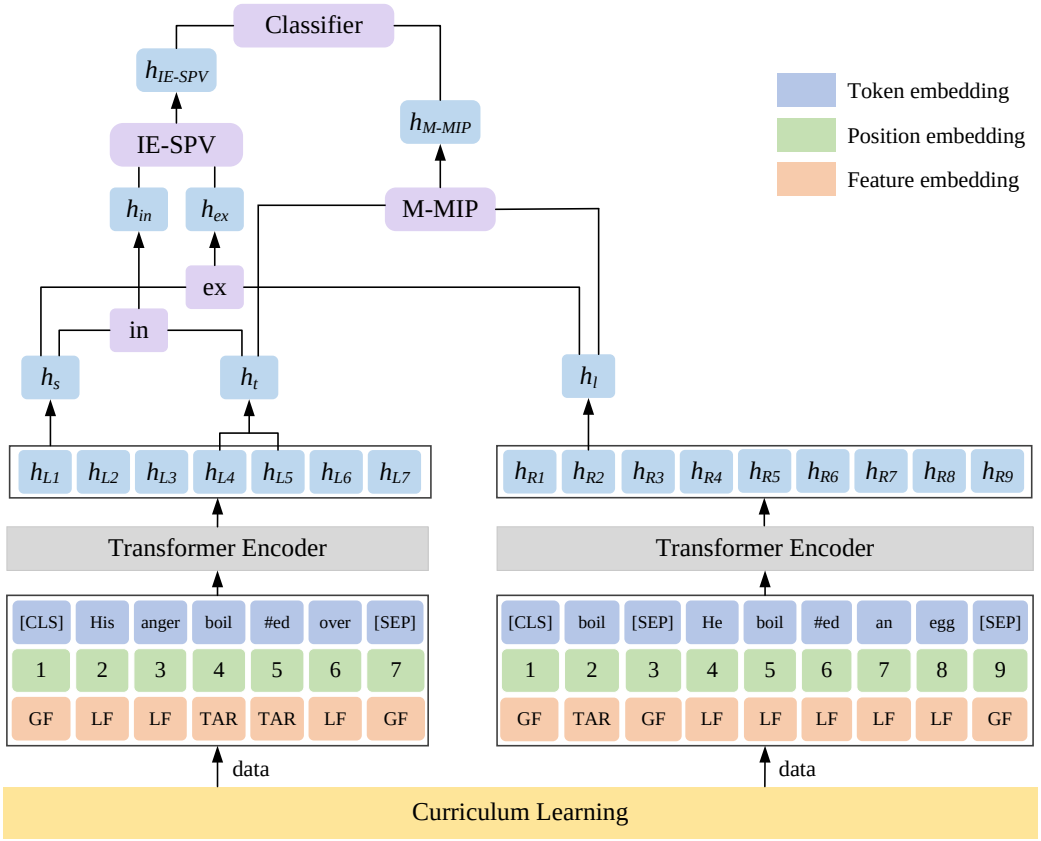

Figure 1: Structrues of MiceCL.

semantic difference between the literal meaning of the target word and the words in the sentence. By combining h_{in} and h_{ex} as h_{SSPV} to reflect the semantic difference between the target word and the context, the model can better identify whether the target word is reasonable in the context, so as to judge the metaphor of the target word.

The sentence internal difference h_{in} is calculated as follows, this readout method can better find out the difference between the two representations while retaining the original information(Zhang and Liu, 2022).

$$h_{in} = W_{in}^T[h_s; h_t; |h_s - h_t|; h_s * h_t] + b_{in} \quad (5)$$

Where W_{in} and b_{in} are the weights and biases of the in-layer (internal layer). The ex-layer (external layer) compares the sentence vector h_s and the literal meaning vector h_l of the target word to represent the sentence external difference h_{ex}, that is, the difference between the literal meaning of the target word and the words of the sentence. We implement ex-layer using a linear transformation:

$$h_{ex} = W_{ex}^T[h_s; h_l; |h_s - h_l|; h_s * h_l] + b_{ex} \quad (6)$$

Where W_{ex} and b_{ex} are the weights and biases of the ex-layer. We combine h_{in} and h_{ex} to represent h_{IE-SPV} :

$$h_{IE-SPV} = concat(h_{in}, h_{ex}) \quad (7)$$

3.1.3 M-MIP

The basic idea of MIP is to identify the metaphoricity of a target word by detecting the difference between its contextual meaning and its literal meaning. The current mainstream methods use the contextual meaning vector of the target word and the literal meaning vector of the target word as the input of the fully connected layer(Choi et al., 2021; Zhang and Liu, 2022) to learn the difference. However, although this approach can learn more features, it will greatly increase the complexity of the model and easily learn more noisy data, making it difficult for the model to learn even basic differences.

Therefore, we propose a new MIP module based on the MIP language rules, which we call the M-MIP (Multiple Metaphor Identification Procedure). In addition to using fully connected layers, we introduce the Manhattan distance between the contextual meaning vector and the literal meaning vector of the target word as the similarity vector h_{sim}. By connecting the difference vector learned by the

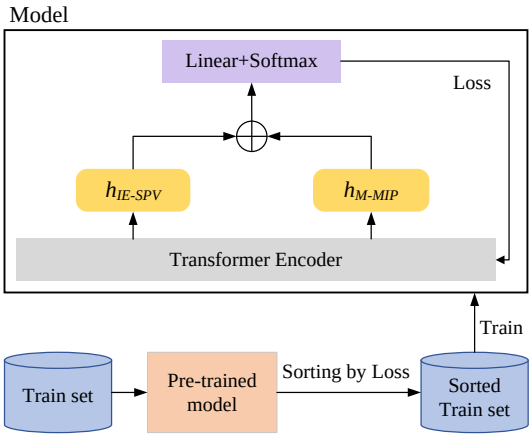

Figure 2: Structures of Curriculum Learning.

fully connected layer with the similarity vector, the model can ensure that it learns a certain number of complex differences while learning the basic differences, so as to better identify the difference between the contextual meaning and the literal meaning of the target word, and judge the metaphor of the target word.

h_{M-MIP} is calculated as follows.

$$h_{M-MIP} = concat(W^T_{M-MIP}[h_t; h_l; |h_t - h_l|; h_t * h_l] + b_{M-MIP}, h_{sim}) \quad (8)$$

Where W_{M-MIP} and b_{M-MIP} are the weights and biases of the M-MIP layer, and the similarity vector h_{sim} is calculated as follows.

$$h_{sim} = \sum_{i=1}^{H} |h_{ti} - h_{li}| \quad (9)$$

Where H is the dimension size of the hidden layer. We combine h_{M-MIP} and h_{IE-SPV} to determine whether the target word is metaphorical or not:

$$\hat{y} = \sigma(W^T[h_{M-MIP}; h_{IE-SPV}] + b) \quad (10)$$

Where W and b are the weights and parameters, σ is a softmax function, and $\hat{y}$ represents the predicted label distribution. Finally, we adopt the cross-entropy loss as the loss function:

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N} w_{yi} y_i log(\hat{y}_i) \quad (11)$$

Where N is the number of training examples, y_i and $\hat{y}_i$ represent the true label and predicted label distribution of the i_{th} example respectively, and w_{yi} is the class weight, which is used to alleviate the data imbalance problem.

3.2 Curriculum Learning

In CLCL(Zhou et al., 2023), curriculum learning is introduced and contrastive loss is used as the difficulty evaluation methods. However, because the model is trained according to the contrastive target and classification target, the impact of only considering contrastive loss is one-sided and cannot fully reflect the difficulty of a sentence. At the same time, because the method of manual evaluation of difficulty relies too much on expert knowledge, and even if the method is logically valid, it is not necessarily applicable to the machine. Certain sentences may be seen as "easy" or "difficult" for a human, but not necessarily the same for a machine. Therefore, instead of manual evaluation, we chose to adopt a framework that automatically measures the difficulty of the sentence. Specifically, we use the training loss of a pre-trained model as the measure of sentence difficulty. As a classification task, we use cross-entropy loss to measure the sentence difficulty:

$$\begin{aligned} d_M(Y_i) &= CTS(Y_i; M) \\ &= -\frac{1}{N}\sum_{i=1}^{N} w_{yi} y_i log(\hat{y}_i) \end{aligned} \quad (12)$$

Where M is the pre-trained model and $d_M(Y_i)$ is the difficulty of Y_i. N is the number of training examples, y_i and $\hat{y}_i$ represent the true label and predicted label distribution of the i_{th} example, respectively, and w_{yi} is the class weight, which is used to alleviate the data imbalance problem.

Algorithm 1: MiceCL

Input: Dataset $\mathbb{P} = \{Y_i\}_{i=1}^{K}$, Pretrained Model M_0, Model M and number of epochs N

Output: Fine-tuned Model M*

```
D_0 = CTS(ℙ, M_0);
Sort ℙ based on each difficulty level in D_0, resulting in a re-arranged ℙ_0;
start = 0.5, speed = (1 − start) * 3/(2N);
for n=1; n<=N do
    percent = min(1, start + speed * n);
    ℙ_n = ℙ_0[: len(ℙ_0) * percent];
    M_n ← TRAIN(ℙ_n);
end
return M* = M_N
```

As shown in Figure 2, we use the pre-trained model to measure the difficulty of the sentences, once we determine the difficulty of the sentences,

Dataset	#Tar	#M	#Sent	#Len
VUA ALL$_{tr}$	116,622	11.19	6,323	18.4
VUA ALL$_{dev}$	38,628	11.62	1,550	24.9
VUA ALL$_{te}$	50,175	12.44	2,694	18.6
VUA Verb$_{tr}$	15,516	27.90	7,479	20.2
VUA Verb$_{dev}$	1,724	26.91	1,541	25.0
VUA Verb$_{te}$	5,783	29.98	2,694	18.6
MOH-X	647	48.69	647	8.0
TroFi	3,737	43.54	3,737	28.3

Table 1: Datasets information. **#Tar**: Number of target words. **#M**: Percentage of metaphors. **#Sent**: Number of sentences. **#Len**: Average sentence length.

we rearrange them in order from easy to difficult and feed them into the model for training, fine-tuning the model based on the training loss. Since the initial ability of the model is weak and will gradually increase over time, we set the initial amount of training data to half of the original amount of data and gradually increase it as the number of epochs increases. In order to better evaluate the effectiveness of the model, we do not make changes to the development and test sets, but adopt the original development and test sets. Through this module, we can ensure that the model will not deal with complex data when its ability is weak, avoid wasting limited data and save computing resources, thus solving the problem of data sparsity and improving the ability and training efficiency of the model. The specific training strategy is shown in Algorithm 1.

4 Experiment

4.1 Datasets

Following previous works on metaphor detection, we conduct experiments on four widely used public datasets: (1) **VUA ALL**(Steen, 2010); (2) **VUA Verb**(Steen, 2010); (3) **MOH-X**(Mohammad et al., 2016); (4) **TroFi**(Birke and Sarkar, 2006). The statistics of the datasets are shown in Table 1, which was summarized by Zhang and Liu (2022).

4.2 Baselines

RNN_ELMo(Gao et al., 2018) and **RNN_BERT** (Devlin et al., 2019): They combine ELMo (or BERT) and GloVe's embeddings to represent a word and use BiLSTM as the base framework.
RNN_HG and **RNN_MHCA**(Mao et al., 2019): RNN_HG uses MIP to compare the differences between the literal and contextual meanings of the target words, which are represented by GloVe and ELMo embeddings, respectively. RNN_MHCA compares the differences between them based on SPV, and uses the multi-head attention mechanism.
MUL_GCN(Le et al., 2020): MUL_GCN uses a multi-task learning framework for metaphor detection and semantic disambiguation.
MelBERT(Choi et al., 2021): RoBERTa based model using both SPV and MIP architectures for metaphor detection.
MrBERT(Song et al., 2021): Treat the metaphor detection task as a relation classification task with relation embeddings as the input to BERT.
MisNet(Zhang and Liu, 2022): RoBERTa based model that uses both SPV and MIP structures for metaphor detection. Different from MelBERT, it improves the representation method of the literal meaning of the target word.
CLCL(Zhou et al., 2023): RoBERTa based model introduces curriculum learning and contrastive learning for metaphor detection on the basis of MelBERT, where curriculum learning adopts the method of manual evaluation of difficulty.

4.3 Experimental Settings

In the experiments, we use RoBERTa(Liu et al., 2019) provided by HuggingFace as the encoder. In the Curriculum learning, the pre-trained model we use for measuring the difficulty is MisNet(Zhang and Liu, 2022). The batch size is 64 with a 1e-5 learning rate. We trained for 10 epochs with a learning rate warmup. All experiments were conducted on a single NVIDIA RTX 3090 GPU. More Details can be found in the appendix A.

5 Results and Analysis

5.1 Overall Results

Table 2 displays the comparative results of MiceCL in contrast to other baselines across VUA ALL, VUA Verb, and MOH-X datasets. Our implementation reveals the remarkable performance achieved by MiceCL. Comparing it to the state-of-the-art model CLCL, MiceCL outperforms it with an F1 score of 0.8 while also achieving the highest accuracy and recall scores. This clearly demonstrates our model's proficiency in predicting complex metaphor usage. MiceCL achieves the best results despite the fact that it does not use POS tags. This shows that IE-SPV and M-MIP modules can make good use of the context and various semantic information, and can correctly judge the semantics

Model	VUA ALL				VUA Verb				MOH-X			
	Acc	P	R	F1	Acc	P	R	F1	Acc	P	R	F1
RNN_ELMo(2018)	93.1	71.6	73.6	72.6	81.4	68.2	71.3	69.7	77.2	79.1	73.5	75.6
RNN_BERT(2019)	92.9	71.5	71.9	71.7	80.7	66.7	71.5	69.0	78.1	75.1	81.8	78.2
RNN_HG(2019)	93.6	71.8	76.3	74.0	82.1	69.3	72.3	70.8	79.7	79.7	79.8	79.8
RNN_MHCA(2019)	93.8	73.0	75.7	74.3	81.8	66.3	75.2	70.5	79.8	77.5	83.1	80.0
MUL_GCN(2020)	93.8	74.8	75.5	75.1	83.2	72.5	70.9	71.7	79.9	79.7	80.5	79.6
MelBERT†(2021)	94.0	80.5	76.4	78.4	80.7	64.6	**78.8**	71.0	81.6	79.7	82.7	81.1
MrBERT(2021)	94.7	**82.7**	72.5	77.2	**86.4**	**80.8**	71.5	75.9	81.9	80.0	85.1	82.1
MisNet†(2022)	94.7	82.4	73.2	77.5	84.4	77.0	68.3	72.4	83.1	83.2	82.5	82.5
CLCL(2023)	94.5	80.8	76.1	78.4	84.7	74.9	73.9	74.4	84.3	**84.0**	82.7	83.4
MiceCL	**95.0**	81.8	**76.8**	**79.2**	85.5	74.5	78.6	**76.5**	**85.2**	83.2	**87.7**	**85.2**

Table 2: Results on VUA All, VUA Verb, and MOH-X. Best in bold and second best in italic underlined. The † results are reproduced by Zhou et al. (2023)

.

and metaphor usage of the target word without the help of POS tags, which proves the effectiveness of IE-SPV and M-MIP modules.

On the VUA Verb dataset, compared to RNN-based and Transformer-based models, our model shows a significant improvement in F1 score by 7.5 and 5.5, respectively. In comparison to the state-of-the-art model MrBERT, MiceCL outperforms it by 0.6 F1 score, highlighting the model's proficiency in predicting the metaphorical usage of verbs. Notably, MrBERT extracts various relations between the subject and object of verbs, while MiceCL achieves superior results solely through semantic matching methods, underscoring the effectiveness of IE-SPV and M-MIP modules.

On the MOH-X dataset, our model exhibits significant improvements, increasing the F1 score by 9.6 and 4.1 when compared to RNN-based and Transformer-based models, respectively. In contrast to the state-of-the-art model CLCL, MiceCL surpasses it by 1.8 F1 scores while achieving the highest accuracy and recall scores. This demonstrates the exceptional efficacy of MiceCL in predicting common metaphorical usages. It is noteworthy that MOH-X contains the least amount of data, and our model outperforms all others by a considerable margin on this dataset, underscoring the importance of the curriculum learning module. By implementing curriculum learning, our model effectively utilizes the limited dataset, providing a robust solution to the problem of data sparsity.

Model	TroFi(Zero-shot)			
	Acc	P	R	F1
MelBERT	-	53.4	74.1	62.0
MrBERT	61.1	53.8	**75.0**	62.7
MiceCL	**61.5**	**54.2**	**75.0**	**62.9**

Table 3: Zero-shot transfer results on TroFi dataset.

5.2 Zero-shot transfer on TroFi

We assess the cross-dataset zero-shot learning capability of our model by training it on the VUA-ALL dataset and subsequently testing it on the TroFi dataset. While this presents a challenging task, it serves as a vital metric for evaluating a model's ability to generalize beyond its training data. As illustrated in Table 3, MiceCL outperforms the state-of-the-art MrBERT by 0.4 points in accuracy, surpasses the state-of-the-art in precision by 0.4 points, matches the state-of-the-art in recall, and exceeds the state-of-the-art in F1 score by 0.2 points. The experimental results demonstrate that our model attains the best performance across all metrics, affirming its strong generalization capabilities and applicability across diverse datasets, not limited to a specific dataset.

5.3 Ablation Study

To investigate the impact of various components in our approach (namely IE-SPV, M-MIP, and curriculum learning), we examined the results of variants without curriculum learning (-CL), without IE-SPV (-in), and without M-MIP (-sim) trained with the

Ablation	Acc	P	R	F1
-CL	66.9	47.0	**80.4**	59.4
-in	**85.8**	**76.4**	76.1	76.3
-sim	85.7	76.2	76.3	76.2
MiceCL	85.5	74.6	78.4	**76.5**

Table 4: Effectiveness study on VUA Verb dataset.

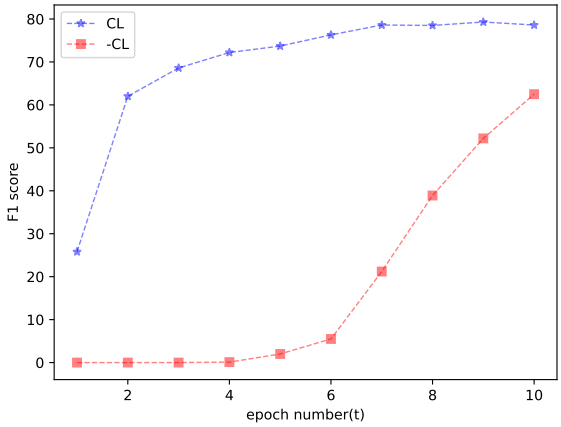

Figure 3: Visualization of convergence rates of different structures.

same hyperparameters on the VUA Verb dataset. As demonstrated in Table 4, all three variants performed less effectively than the full model. When the curriculum learning module was absent, the model's performance was notably diminished, resulting in an 18.6-point drop in accuracy and a 17.1-point decrease in F1 score. This underscores the critical role played by the curriculum learning module. In cases where the IE-SPV or M-MIP modules were missing, although accuracy increased, the F1 score dropped by 0.2 and 0.3, respectively. This suggests that the IE-SPV and M-MIP modules also contribute significantly to metaphor recognition. Consequently, it is only through their combined utilization that they complement and enhance each other, resulting in the best overall performance.

5.4 Analysis on Curriculum Learning

As evident from Table 4, curriculum learning exerts the most substantial influence on the model's results. To delve into the specific impact of curriculum learning, we separately calculated the F1 value for each epoch to compare the model's performance with and without curriculum learning. The results are depicted in Figure 3. We observed that in the absence of curriculum learning, the model's convergence is notably sluggish, struggling to acquire

MiceCL	- CL	- in	- sim	Sentence
✓				And if your father is currently taking your side of things worth sounding him out as to how *far* he 'll take.
✓				A proportion of both boys and girls *at* ego-identity achievement stage will choose science.
✓		✓	✓	A conception of *autonomy* which depends upon group membership displays its own contradiction.
✓	✓		✓	It will take me bleeding years to reach my *goal*.
✓	✓	✓		Shall we *take* photographs?
				She *bought* it.
				Except that the Tysons don't *gamble*.

Table 5: Examples of incorrect samples for MiceCL on VUA ALL. The metaphorical words in the sentence are in red italicized. ✓marks correct model prediction.

valuable features during the initial phases of training. It achieves an F1 score of 59.4 after 10 epochs of training, while the model with curriculum learning has already achieved an F1 score of over 60 by the second epoch. The introduction of curriculum learning leads to a significant acceleration in the model's convergence, effectively reducing the computational resource requirements. Simultaneously, by allowing the model to avoid processing complex data during its early stages, it prevents the wastage of limited data and efficiently resolves the problem of data sparsity.

5.5 Case Study

Table 5 shows the results of the case analysis. The top two examples prove that MiceCL can make full use of the context information and can better identify the use of metaphor than other models when the context information is rich. The third example proves that without adopting curriculum learning, it is difficult to deal with the data sparsity problem and solve the metaphorical usage of unusual words. The fourth and fifth examples prove that if IE-SPV and M-MIP are not adopted, the model may struggle to learn very basic differences and make mistakes in the recognition of some very basic metaphorical usages. The sixth and seventh examples are not recognized by all the models, and such cases will have great difficulties in recognizing metaphors due to the lack of context information, which can be left for future work to solve.

6 Conclusion

In this paper, we propose a novel metaphor detection model called MiceCL. The proposed model includes two modules, M-MIP and IE-SPV, to identify metaphors by effectively capturing the basic and complex differences between target words and context information. In addition, we introduce curriculum learning to automatically measure the difficulty with a pre-trained model to re-arrange the training order, eliminate the disadvantages of manual evaluation of difficulty, and largely solve the problem of data sparsity. We evaluate our model on four datasets, and the effect shows a significant improvement over the strong baselines. We provide detailed ablation experiments to demonstrate the effectiveness of our approach.

Limitations

Our curriculum learning framework evaluates the difficulty of the data and re-arranges the training order before training, after which the training order is fixed. In fact, with the deepening of training, the ability of the model will continue to change, so the curriculum learning framework can be improved to continuously re-arrange the training order, so as to improve the flexibility of the model. In addition, we assume that the capacity of the model grows linearly, so the size of the training set increases linearly, but in reality the capacity of the model does not necessarily grow linearly with time. We leave these to a future study.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (No.62302121).

References

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In Proceedings of the 26th Annual International Conference on Machine Learning, ICML '09, page 41–48, New York, NY, USA. Association for Computing Machinery.

Julia Birke and Anoop Sarkar. 2006. A clustering approach for nearly unsupervised recognition of nonliteral language. In 11th Conference of the European Chapter of the Association for Computational Linguistics, pages 329–336, Trento, Italy. Association for Computational Linguistics.

George Aaron Broadwell, Umit Boz, Ignacio Cases, Tomek Strzalkowski, Laurie Feldman, Sarah Taylor, Samira Shaikh, Ting Liu, Kit Cho, and Nick Webb. 2013. Using imageability and topic chaining to locate metaphors in linguistic corpora. In Proceedings of the 6th International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction, SBP'13, page 102–110, Berlin, Heidelberg. Springer-Verlag.

Luana Bulat, Stephen Clark, and Ekaterina Shutova. 2017. Modelling metaphor with attribute-based semantics. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers, pages 523–528, Valencia, Spain. Association for Computational Linguistics.

Minjin Choi, Sunkyung Lee, Eunseong Choi, Heesoo Park, Junhyuk Lee, Dongwon Lee, and Jongwuk Lee. 2021. MelBERT: Metaphor detection via contextualized late interaction using metaphorical identification theories. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 1763–1773, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Ge Gao, Eunsol Choi, Yejin Choi, and Luke Zettlemoyer. 2018. Neural metaphor detection in context. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 607–613, Brussels, Belgium. Association for Computational Linguistics.

Hongyu Gong, Kshitij Gupta, Akriti Jain, and Suma Bhat. 2020. IlliniMet: Illinois system for metaphor detection with contextual and linguistic information. In Proceedings of the Second Workshop on Figurative Language Processing, pages 146–153, Online. Association for Computational Linguistics.

A. Graves and J. Schmidhuber. 2005. Framewise phoneme classification with bidirectional lstm networks. In Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005., volume 4, pages 2047–2052 vol. 4.

Pragglejaz Group. 2007. Mip: A method for identifying metaphorically used words in discourse. Metaphor and Symbol, 22(1):1–39.

G. Lakoff and M. Johnson. 2008. Metaphors We Live By. University of Chicago Press.

Duong Le, My Thai, and Thien Nguyen. 2020. Multi-task learning for metaphor detection with graph convolutional neural networks and word sense disambiguation. Proceedings of the AAAI Conference on Artificial Intelligence, 34(05):8139–8146.

Qing Li, Siyuan Huang, Yining Hong, and Song-Chun Zhu. 2020. A competence-aware curriculum for visual concepts learning via question answering. In Computer Vision – ECCV 2020, pages 141–157, Cham. Springer International Publishing.

Xuebo Liu, Houtim Lai, Derek F. Wong, and Lidia S. Chao. 2020. Norm-based curriculum learning for neural machine translation. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 427–436, Online. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. arXiv e-prints, page arXiv:1907.11692.

Rui Mao, Chenghua Lin, and Frank Guerin. 2019. End-to-end sequential metaphor identification inspired by linguistic theories. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 3888–3898, Florence, Italy. Association for Computational Linguistics.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2, NIPS'13, pages 3111–3119, Red Hook, NY, USA. Curran Associates Inc.

Saif Mohammad, Ekaterina Shutova, and Peter Turney. 2016. Metaphor as a medium for emotion: An empirical study. In Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics, pages 23–33, Berlin, Germany. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Emmanouil Antonios Platanios, Otilia Stretcu, Graham Neubig, Barnabas Poczos, and Tom Mitchell. 2019. Competence-based curriculum learning for neural machine translation. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 1162–1172, Minneapolis, Minnesota. Association for Computational Linguistics.

Marek Rei, Luana Bulat, Douwe Kiela, and Ekaterina Shutova. 2017. Grasping the finer point: A supervised similarity network for metaphor detection. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 1537–1546, Copenhagen, Denmark. Association for Computational Linguistics.

Ekaterina Shutova, Douwe Kiela, and Jean Maillard. 2016. Black holes and white rabbits: Metaphor identification with visual features. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 160–170, San Diego, California. Association for Computational Linguistics.

Ekaterina Shutova and Lin Sun. 2013. Unsupervised metaphor identification using hierarchical graph factorization clustering. In Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 978–988, Atlanta, Georgia. Association for Computational Linguistics.

Ekaterina Shutova, Lin Sun, and Anna Korhonen. 2010. Metaphor identification using verb and noun clustering. In Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010), pages 1002–1010, Beijing, China. Coling 2010 Organizing Committee.

Wei Song, Shuhui Zhou, Ruiji Fu, Ting Liu, and Lizhen Liu. 2021. Verb metaphor detection via contextual relation learning. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 4240–4251, Online. Association for Computational Linguistics.

Gerard Steen. 2010. A method for linguistic metaphor identification : from mip to mipvu.

Chuandong Su, Fumiyo Fukumoto, Xiaoxi Huang, Jiyi Li, Rongbo Wang, and Zhiqun Chen. 2020. DeepMet: A reading comprehension paradigm for token-level metaphor detection. In Proceedings of the Second Workshop on Figurative Language Processing, pages 30–39, Online. Association for Computational Linguistics.

Yulia Tsvetkov, Leonid Boytsov, Anatole Gershman, Eric Nyberg, and Chris Dyer. 2014. Metaphor detection with cross-lingual model transfer. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 248–258, Baltimore, Maryland. Association for Computational Linguistics.

Peter Turney, Yair Neuman, Dan Assaf, and Yohai Cohen. 2011. Literal and metaphorical sense identification through concrete and abstract context. In Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, pages 680–690, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS'17, page 6000–6010, Red Hook, NY, USA. Curran Associates Inc.

Daphna Weinshall, Gad Cohen, and Dan Amir. 2018. Curriculum learning by transfer learning: Theory and experiments with deep networks. In Proceedings of the 35th International Conference on Machine Learning, volume 80 of Proceedings of Machine Learning Research, pages 5238–5246. PMLR.

Yorick Wilks. 1978. Making preferences more active. Artificial Intelligence, 11(3):197–223.

Chuhan Wu, Fangzhao Wu, Yubo Chen, Sixing Wu, Zhigang Yuan, and Yongfeng Huang. 2018. Neural metaphor detecting with CNN-LSTM model. In Proceedings of the Workshop on Figurative Language Processing, pages 110–114, New Orleans, Louisiana. Association for Computational Linguistics.

Mingliang Zhang, Fandong Meng, Yunhai Tong, and Jie Zhou. 2021. Competence-based curriculum learning for multilingual machine translation. In Findings of the Association for Computational Linguistics: EMNLP 2021, pages 2481–2493, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Shenglong Zhang and Ying Liu. 2022. Metaphor detection via linguistics enhanced Siamese network. In Proceedings of the 29th International Conference on Computational Linguistics, pages 4149–4159, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Jianing Zhou, Ziheng Zeng, and Suma Bhat. 2023. CLCL: Non-compositional expression detection with contrastive learning and curriculum learning. In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 730–743, Toronto, Canada. Association for Computational Linguistics.

Lei Zhou, Liang Ding, Kevin Duh, Shinji Watanabe, Ryohei Sasano, and Koichi Takeda. 2021. Self-guided curriculum learning for neural machine translation. In Proceedings of the 18th International Conference on Spoken Language Translation (IWSLT 2021), pages 206–214, Bangkok, Thailand (online). Association for Computational Linguistics.

A Experimental Details

All experiments were conducted on a single NVIDIA RTX 3090 GPU.

A.1 Hyperparameter Choices

VUA ALL: For VUA ALL dataset, the learning rate is 1e-5 with learning rate warmup, the number of epochs is 10, and the batch size is 64.
VUA Verb: For VUA Verb, the learning rate is 1e-5 and the learning rate warmup is used, the number of epochs is 6, and the batch size is 64.
MOH-X: For MOH-X, the learning rate is fixed to 1e-5, the number of epochs is 15, and the batch size is 64.
TroFi: For TroFi, we only use it for zero-shot evaluation, where the model is trained on VUA ALL and tested on TroFi.

A.2 Average Runtime

When using a single NVIDIA RTX 3090 GPU, the process of measuring sentence difficulty takes approximately five minutes, and training for one epoch requires about fifteen minutes.