

Nominal Metaphor Generation with Multitask Learning

Yucheng Li¹, Chenghua Lin^{2*}, Frank Guerin¹

¹ Department of Computer Science, University of Surrey, UK

{yucheng.li, f.guerin}@surrey.ac.uk

² Department of Computer Science, University of Sheffield, UK

c.lin@sheffield.ac.uk

Abstract

Metaphor generation is a challenging task which can impact many downstream tasks such as improving user satisfaction with dialogue systems and story generation. This paper tackles the problem of Chinese nominal metaphor generation by introducing a multitask metaphor generation framework with self-training and metaphor identification mechanisms. Self-training addresses the data scarcity issue of metaphor datasets. That is, instead of solely relying on labelled metaphor datasets which are usually small in size, self-training helps identify potential metaphors from a large-scale unlabelled corpus for metaphor generation. The metaphor weighting mechanism enables our model to focus on the metaphor-related parts of the input (e.g., the comparison of the metaphor and comparator) during model learning and thus improves the metaphoricity of the generated metaphors. Our model is trained on an annotated corpus consisting of 6.3k sentences that contain diverse metaphorical expressions. Experimental results show that our model is able to generate metaphors with better readability and creativity compared to the baseline models, even in the situation where training data is insufficient.

1 Introduction

Metaphor is commonly used in human language as an effective communication device. Typically, metaphors compare a concept or an object to another with the intent to make the expression more vivid, or to make unfamiliar things easier to understand (Paul, 1970).

According to linguistic studies of Chinese, metaphors are particularly important in Chinese as metaphor is the dominant figurative language in Chinese (Wang, 2004). As shown in Table 1, there are different types of Chinese metaphors. *Nominal* Metaphors (NMs), also known as 比喻 in

1. 百合花好似一瓶香水. Lilies smells like a bottle of perfume.	<i>Nominal</i>
2. 他很擅长编织人际关系. He is good at weaving relationship.	<i>Verb</i>
3. 银行股跳水 Bank stock dive.	<i>Personification</i>
4. 他可以像大厨一样烹饪 He can cook like a pro.	<i>Non</i>

Table 1: Examples of different types of Chinese metaphor: *nominal* metaphor, *verb* metaphor, and *personification* metaphors. Metaphorical words are bold.

Chinese, are figures of speech associating a noun with another noun through a comparator such as 像, 是, 变成 (equivalent to *like*, *be*, *become* in English). Wang (2004) claims that the nominal metaphor requires the comparison to be drawn from objects different in nature. Therefore, even though the fourth example in the table uses a classic comparator “like”, it does not make it a metaphor as it compares a person to another person. *Verb* metaphors are metaphors whose verbs are used metaphorically. The verb metaphor shown in Table 1 uses *weaving*, a verb which is usually related to cloth or loom, to describe human relationships. The third type of Chinese metaphor is *personification* (also known as 拟人 in Chinese), which treats objects as human and can act like humans.

Previous efforts on metaphor generation demonstrate the task can bring benefits to a wide range of NLG downstream tasks. Glucksberg (1989) suggested that *verb* metaphors are important to an engaging conversation. Zhou (2020) showed machine-generated NMs are effective in stimulating user interest in communicating with chatbots. Chakrabarty et al. (2020, 2021) conducted human evaluations comparing literal expressions from machine-generated stories and poems with

*Corresponding author

machine-generated metaphors and found users prefer the text with metaphors.

In this paper, we mainly focus on generating the *nominal* metaphors. The generation of the NMs is defined as follows: given the subject of the metaphor, i.e., “*Lilies*” in the first example of Table 1, generate a comparison containing the comparator and the object of the comparison, i.e., “*like*” and “*a bottle of perfume*” in the example, respectively. There are two main challenges to the Chinese metaphor generation task. The first issue is the annotated corpus. Existing Chinese metaphor corpora are not large enough to power current data-driven text generation approaches. Second, the auto-regressive fashion language modelling is ineffective for learning metaphor generation. Because a metaphor can be hidden in a very long sentence, the generative model tends to learn the entire sentence sequence rather than focusing on the metaphorical part of the input.

To address the aforementioned challenges, we propose a novel neural metaphor generation model that requires only limited labelled metaphor data for model training. This is achieved by a multi-task framework which jointly performs novel self-training and metaphor weighting mechanisms. First, to tackle the scarcity issue of metaphor datasets, we employ self-training to leverage additional unlabelled data to improve the metaphor generation performance. Self-training consists of three main steps: (1) train a teacher model on labelled training data; (2) detect potential metaphors in the unlabelled corpus; and (3) train a student model on the combination of the labelled as well as newly identified metaphors from the unlabelled data. Second, we propose to employ metaphor identification to reveal metaphor-related parts of the input. This permits our model to focus on the metaphor-related parts of the entire input sentence via assigning higher weights to metaphor-related content. Introducing metaphor identification not only improves the efficiency of the model training process, but also improves the metaphoricity of the generated metaphors.

As there are limited data available for nominal metaphor generation, we collect and annotate two corpora for our model training, namely, Chinese Metaphor Corpus (CMC) and Chinese Literature Corpus (CLC). CMC contains 2.7k metaphor examples and 3.5k literal examples, which can be used for both metaphor detection and gen-

eration. CLC is a large-scale unlabelled Chinese literature corpus, which can be leveraged by our self-training algorithm for identifying additional high quality labelled metaphors. We conduct both automatic and human evaluation to evaluate our model’s performance in metaphor generation. Experimental results show that our model is able to generate metaphors with better readability and creativity compared to the baseline models, even in the situation where training data is insufficient. Source code and data can be found in https://github.com/liyucheng09/Metaphor_Generator.

2 Related Work

Prior works on computational processing of metaphors can generally be classified into detection, interpretation and generation tasks.

2.1 Detection and Interpretation of Metaphors

Krishnakumaran and Zhu (2007) exploit the absence of a hyponymy relation between subject and object to identify metaphorical utterances. Shlomo and Last (2015) propose a random forest-based classifier for NM identification using both conceptual features such as abstractness and semantic relatedness such as domain corpus frequency. Su et al. (2016) follow the idea of hyponymy relationship absence from (Krishnakumaran and Zhu, 2007) and implement it using cosine distance between pre-trained word2vec embeddings of the source and target concepts. Liu et al. (2018); Zeng et al. (2020) tackle Chinese simile detection by designing a multi-task framework and a local attention mechanism. Here, simile is a type of NM, which uses direct comparator such as “*like*” and “*as*”. Su et al. (2016, 2017) focus on *Nominal* and *verb* metaphor interpretation and perform experiments on English and Chinese metaphors. They extract properties of both the subject and object of the metaphors from WordNet and use pre-trained word2vec embeddings to identify related properties shared by the compared objects/concepts pair.

2.2 Generation of Metaphors

Despite the benefits that metaphor generation can bring to many NLG tasks, works on this task are still relatively sparse. Early works for metaphor generation often rely on templates. Terai and Nakagawa (2010) compute the relatedness between

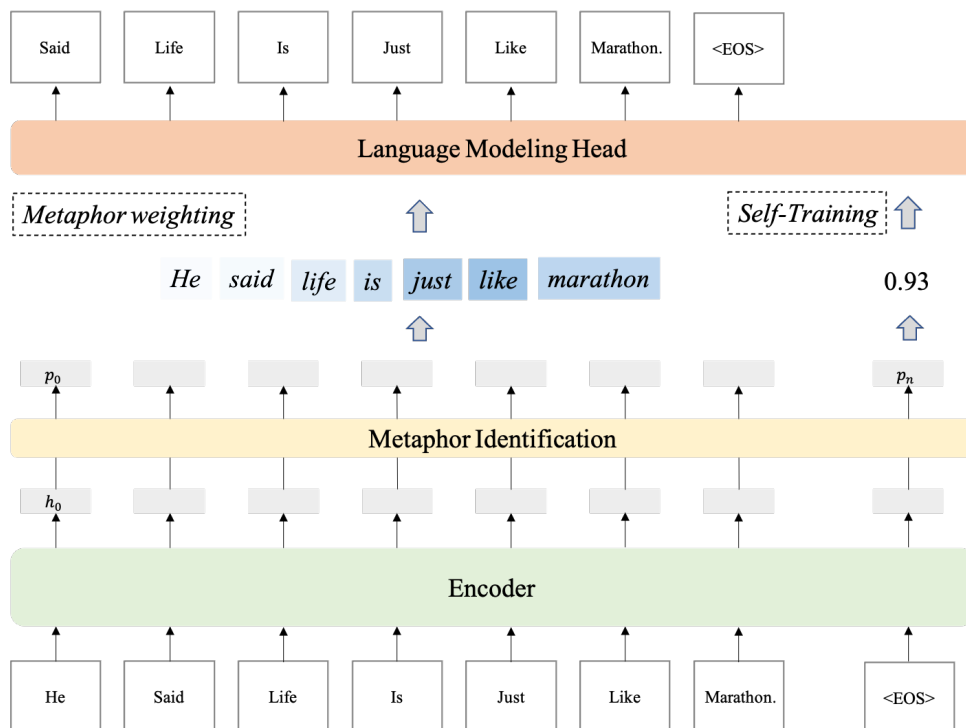


Figure 1: The overall framework of our method. The metaphor identification module performs self-training and metaphorical word identification.

concepts with computational language analysis and select candidates to fill metaphor templates, e.g., “A is like B”. Veale (2016) uses a knowledge-base to generate *XYZ* style metaphors such as “Bruce Wayne is the Donald Trump of Gotham City”. Zhou (2020) not only chooses candidate concept pairs by word embedding similarity to fill the template but also chooses appropriate comparators to link the concept pair. Chakrabarty et al. (2020) introduce a neural style transfer approach for simile generation, which fine-tunes a pre-trained sequence-to-sequence model on a literal-simile parallel dataset. Nevertheless, previous template-based approaches heavily constrain the diversity of generated metaphors, and both template methods and neural methods produce metaphors with a relatively simple structure.

2.3 Metaphor Corpora

Annotating metaphors can be quite challenging as cognitive scientists and psychologists define a variety of theories for metaphor. One popular approach to annotate metaphors in open text is the Metaphor Identification Procedure (MIP) introduced by Steen (2010). Their study also releases the first metaphor dataset named VUAMC, which focuses on *verb* metaphors and contains around 7k

metaphor examples. VUAMC is widely used in metaphor identification and served as a benchmark in the first and second ACL workshop on figurative languages (Leong et al., 2018, 2020). Chakrabarty et al. (2020) crawl web content and construct the first English simile corpus to train their metaphor generator. Liu et al. (2018) release a small Chinese metaphor corpus with 120 examples including both *verb* and *nominal* metaphors. (Zeng et al., 2020) publish a Chinese simile dataset focusing on a specific NM which using the comparator “like”.

3 Methodology

In this section, we provide the technical detail of our proposed framework metaphor generation. The overall model architecture is shown in Figure 1, which includes: (1) the GPT2 model (Radford et al., 2019) and text prediction layer; (2) the metaphor identification module, which identifies potential metaphors from the unlabelled dataset (i.e., via self-training) and emphasises metaphorical words of the input sequence (i.e., via metaphor weighting).

3.1 Text Modeling and Metaphor Identification Module

The Pre-Trained Language Model We employ GPT2, a pre-trained unidirectional transformer lan-

guage model, as our basic encoder. Given a sentence $S = (w_0, \dots, w_n)$, the GPT2 model produces a list of contextualized token embedding (h_0, \dots, h_n) , where h_i is the representation of the i -th input token w_i . Since the GPT2 model is a *unidirectional* language model, the produced contextualized word embedding captures only the information of preceding context. For example, h_i captures information asserted before the i -th token, which is (w_0, \dots, w_i) .

The Text Prediction Layer To predict the output token, we apply a linear layer followed by a softmax function to the contextualized word embedding.

$$P(w_{i+1}|w_1, \dots, w_i) = \text{softmax}(Wh_i + b) \quad (1)$$

where W and b are trainable weight matrix and bias for text generation, respectively; h_i is the contextualised word embedding of the i -th word.

The Metaphor Identification Module The metaphor identification module is used to assign metaphorical probability to *sentences* (used in self-training) or *sub-sentences* (used in metaphor weighting). Specifically, we apply a linear layer plus a softmax layer on the contextualized embedding to compute the metaphorical probability. Formally, after obtaining word representations (h_0, h_1, \dots, h_n) by GPT2, we compute the metaphor probability as follow:

$$p_i = \text{softmax}(W_m h_i + b_m) \quad (2)$$

where W_m and b_m are the trainable weight matrix and bias for metaphor identification, and p_i is the metaphor probability of the sub-sentence w_0, \dots, w_i (or the whole sentence if $i = n$). Because h_i captures the contextual information preceding w_i , p_i indicates whether the sub-sentence w_0, \dots, w_i contains metaphorical expression.

3.2 Self-Training

Self-training is an effective approach to make use of additional data to enhance deep learning model performance. It has shown significant progress in many deep learning tasks, such as machine translation (He et al., 2019), speech recognition (Parthasarathi and Strom, 2019), and image classification (Xie et al., 2020). As shown in (He et al., 2019), self-training provides significant performance gains in sequence generation when the supervised corpus is relatively small. To ensure

Algorithm 1: Classic Self-Training

```

1 Train a base model  $f$  on  $U = \{x_i, y_i\}$ ;
2 repeat
3   Apply  $f$  to the unlabeled instance  $C$ ;
4   Select a subset  $S \subset \{(x, f(x)) | x \in C\}$ 
   ;
5   Train a new model  $f'$  on  $S \cup U$ ;
6 until convergence or maximum iterations
   are reached;
```

generation fluency and diversity, we employ self-training to train our metaphor generator.

The classic self-training procedure is shown in Algorithm 1. Classic self-training starts from a teacher model trained with a supervised dataset U . The teacher model is then applied to unlabelled data to obtain the pseudo label. Finally, we train a student model on the syntactic dataset incorporating the pseudo label dataset and the supervised dataset $S \cup U$.

Instead of training a *independent* teacher model, we embed the teacher model in the overall model as an attention mechanism. As shown in Figure 1, each unlabelled instance is weighted by its metaphorical probability during metaphor modelling. We use the metaphor identification module to score each sentence with the probability of being metaphorical. Formally, given an unlabelled sentence $x = (w_0, \dots, w_n)$, the metaphor identification module compute its metaphorical probability p_n .

$$p_n = \text{softmax}(W_m h_n + b_m) \quad (3)$$

where h_n is the representation of w_n , the last token of x . Since h_n captures context information of the entire sentence, we regard p_n as the metaphorical probability of x and use it to weight this training instance.

3.3 Metaphor Weighting

We find that in metaphorical sentences, only partial of the text is metaphor-related(i.e., the comparison between objects/concepts pair and comparator). To verify, we analyse 200 metaphorical sentences randomly sampled from our dataset. We empirically find that for these metaphorical sentences, only 27% of words are metaphor-related. This indicates that it is necessary to encourage our model to focus on metaphor-related parts of input metaphors; otherwise, it might lead to lower training efficiency and sub-optimal performance.

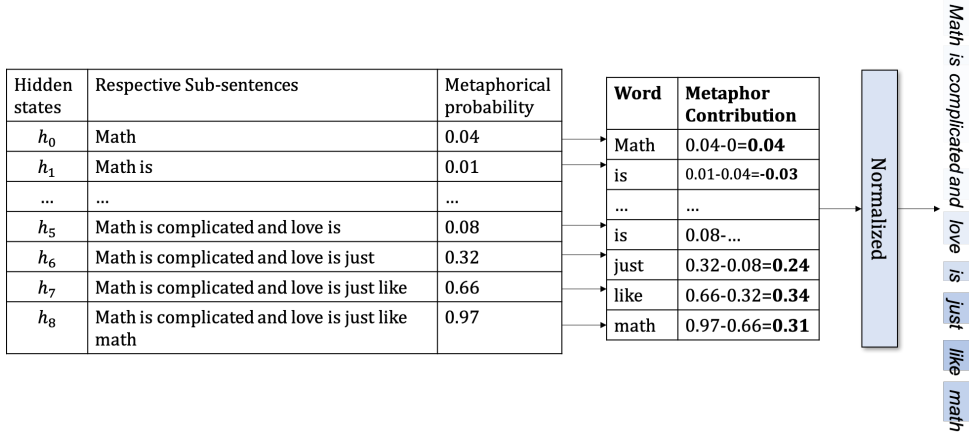


Figure 2: Metaphorical words identification by computing its contribution to metaphor probability.

To tackle this issue, we propose a novel approach based on metaphor detection to identify metaphorical words within sentences. As shown in Figure 2, this is achieved by calculating the contribution to metaphor probability of each word to identify metaphor-related words. Formally, we use I_i to indicate the importance of the i -th token in the metaphor modelling.

$$I_i = p_i - p_{i-1} \quad (4)$$

where p_{i-1}, p_i represents the metaphorical probability of subsentences w_0, \dots, w_{i-1} and w_0, \dots, w_i . If I_i is positive, it indicates the i -th token makes the sub-sentence w_0, \dots, w_i more metaphorical. On the contrary, if I_i is zero or negative, it means this token is irrelevant to metaphorical expression. We weight each training step (i.e., given w_0, \dots, w_{i-1} , predict w_i) with its corresponding I'_i , which is computed as follow:

$$I'_i = \frac{\exp(\max(0, I_i))}{\sum_1^n \exp(\max(0, I_i))} \quad (5)$$

3.4 Training and Inference

Training The training process of *metaphor identification* is as follows. Given a labelled corpus $U = \{(x_i, y_i)\}_{i=1}^N$, where threentance consists of a sentence x and a label y indicating whether x is a metaphorical sentence. We minimize the following loss function:

$$\begin{aligned} L &= - \sum_{x \in U} \log P(\hat{y}|x) \\ &= - \sum_{x \in U} \log(\text{softmax}(W_m h_n + b_m)) \end{aligned} \quad (6)$$

where h_n is the representation of the last token of x .

The training procedure of *metaphor modelling* is as follows. Given an unlabelled dataset $C = \{x_i\}_{i=1}^N$ where each instance is a sentence $x = (w_0, \dots, w_n)$ with a list of words. Each instance is weighted by its metaphorical probability and each token w_i is weighted by I_i , i.e., the metaphorical probability of word w_i . Formally, the loss function of our metaphor modelling is given as follows:

$$L = - \sum_{x \in U \cup C} P(\hat{y}|x) \sum_i^n I'_i \cdot \log P(w_i | w_0, \dots, w_{i-1}) \quad (7)$$

Inference At the inference stage, we regard our model as a normal metaphorical language model to generate metaphors and do not perform metaphor identification. Given a target word w_t , i.e., the subject of the metaphor, we feed the target word concatenated with delimiter as input to our model. Our model produces the next word recurrently until the ENDOFSENTENCE token is generated.

4 Experiment

4.1 dataset

To train our multitask framework, we construct two datasets: a labeled Chinese Metaphor Corpus (CMC) and a large-scale unlabeled Chinese Literature Corpus (CLC).

CMC dataset Existing Chinese metaphor corpus are neither too small (e.g., Su et al. (2016) contains only 120 examples), or focus on a specific comparison, such as Liu et al. (2018) which only consider metaphors with a specific comparator 像 (like). We annotate Chinese Metaphor Corpus (CMC) to cover as many diverse metaphor patterns as possible. CMC contains a total of 2703

	CMC	CLC
# Sentences	6257	6.98M
# Metaphors	2787	-
# literal sentence	3554	-
# tokens	225K	202M
# tokens per sentence	35	29

Table 2: Statistics of CMC and CLC datasets

metaphors and 3554 ordinary sentences, and is used to train our model to detect potential metaphors.

The 6.3k examples are sampled from 47,000 annotated sentences from Chinese literature corpus. We recruited three native Chinese annotators to perform the metaphor annotation task. Before annotation, we provide annotation guidelines to annotators and explain relevant definitions with metaphor examples. The annotation task is to judge whether a sentence is a metaphor. Each sentence is labelled by three annotators and the final label takes the majority of the three labelled results. The annotation agreement rate of CMC is 0.84 based on Krippendorff’s alpha (Krippendorff, 2011) statistics. The statistics of CMC are shown in Table 2, along with some metaphor examples given in Table 3.

CLC dataset In self-training, we need a large-scale corpus to enable the metaphor identification module detecting novel NMs. However, popular Chinese corpora, such as news, Wikipedia, web pages, are not suitable to be used as metaphor resources. Intuitively, literature text might be a promising resource of diverse metaphors. Therefore, we construct a Chinese literature corpus by collecting a large number of essays, novels, and fictions (see details in Appendix A). The statistics of CLC are shown in Table 2.

4.2 Baselines

Chinese metaphor generation is a novel task. We select three general generative models and an English simile generation method as baselines.

RNN: A LSTM based auto-regressive generative model, which consists of three LSTM layers.

SeqGAN: Sequence Generative adversarial network (Yu et al., 2017) with a generator implemented by LSTM network and a discriminator implemented by CNN network. We train this model on CMC to produce Chinese metaphor.

GPT2: The Chinese GPT2 model is fine-tuned on the CMC dataset to produce Chinese metaphors as a baseline model.

Type	Examples
Metaphor	瀑布注入水潭的一刹那,一朵朵白色的一浪一花腾空而起,像溅玉抛珠一般。 At the moment when the waterfall was poured into the pool, a white spray of flowers vacated, like a splash of jade beads.
Not metaphor	泛着银光的大海在身后铺展开来。 The silver-filled sea spread out behind him.

Table 3: Examples of metaphor and not metaphor in CMC.

BART: We construct parallel data (*target word, metaphor*) from CMC and use the paired data to fine-tune a Chinese version BART model (Shao et al., 2021) model.

SCOPE: (Chakrabarty et al., 2020) A SOTA method on English simile generation tasks, which fine-tunes BART model on a large-scale automatically created literal-simile parallel corpus.

4.3 Experiments Setting

We use a pre-trained Chinese GPT2 model¹ to avoid starting training from scratch. Our model is pre-trained on metaphor identification task with CMC for 3 epochs before jointly optimizing the two task-specific loss functions. The implementation of SeqGAN² and the pre-trained Chinese BART model³ can be found in the footnote. Before training the RNN model and SeqGAN models on CMC, we first pre-train the RNN and the generator of SeqGAN on CLC for 50k steps. Hyperparameters not specified are all followed by pytorch default settings. Note that the SCOPE model is designed for English Simile generation and it takes a literal utterance as input. To compare SCOPE results with our method, we first translate the input target word into English using Google Translator, and then translate the generated outputs back to Chinese (details given in Appendix B). In the test stage, we randomly select and feed 200 target words from CMC to all generative models. During

¹<https://huggingface.co/uer/gpt2-chinese-cluecorpussmall>

²<https://github.com/LantaoYu/SeqGAN>

³<https://huggingface.co/fnlp/bart-base-chinese>

Methods	Automatic Eval				Human Eval		
	PPL	Dist-1	Dist-2	Meta	Fluency	Consistency	Creativity
RNN	45.617	.00758	.1564	.955	1.60 (.58)	2.25 (.42)	2.65 (.31)
SeqGAN	89.43	.00336	.0116	.998	3.33 (.51)	3.80 (.46)	1.67 (.34)
GPT2	57.88	.00916	.1154	.981	4.00 (.62)	3.10 (.39)	2.60 (.31)
BART	48.58	.00826	.0971	.978	4.35 (.54)	3.05 (.37)	2.30 (.32)
SCOPE	92.32	.00517	.0673	.910	3.10 (.64)	2.70 (.44)	2.10 (.45)
Our Method	26.79	.01143	.1582	.952	4.55 (.58)	4.23 (.45)	3.80 (.36)
w/o Self-training	62.54	.00674	.0906	.982	3.85 (.54)	3.87 (.42)	2.76 (.38)
w/o Identification	27.58	.01050	.1529	.803	4.50 (.63)	3.91 (.32)	3.41 (.43)

Table 4: Results of automatic metrics and human evaluation. Boldface denotes the best results among our method and baselines. The inter-annotator agreement for human evaluation are shown in parenthesis.

decoding, all beam sizes are set to 12, thus each model generated 12 sentence for each target. In total, 2400 sentences are obtained per model for testing.

4.4 Metrics

Automatic Metrics We use perplexity (PPL) to evaluate the fluency of the generated text, which is calculated by an open source Chinese language model (Zhang et al., 2020). **Dist-1** and **Dist-2** (Li et al., 2016) compute the distinct unigrams and bigrams ratio of generated text, which are used to measure model’s ability to produce diversity outputs. To test the **metaphoricity (Meta)** of generated outputs, we train a RoBERTa-based Chinese metaphor classifier on CMC to compute the ratio of metaphorical utterances in the generated sentences. The accuracy of this classifier is 97.89%, which is robust enough to perform evaluation. We show details of the classifier in Appendix C.

Human Evaluation Due to the creative and delicate usage of metaphor, automatic metrics are not adequate to test the quality of generated outputs. We also perform human evaluation based on the following three criteria: 1) **Fluency** indicates how well the metaphor is formed; whether the expression is grammatical and fluent. 2) **Consistency** indicates whether the metaphor make sense; how well the subject of the metaphor related to the object. 3) **Creativity** scores how creative annotators think the metaphor is. Note that the Creativity judgment is based on annotators’ real-life experience, rather than measuring whether the generated metaphor appears in the training dataset. Three annotators were instructed to rate the three criteria from 1 to 5, where 1 denotes worst and 5 be the

best.

5 Results

5.1 Automatic Evaluation

Results of automatic metrics are shown in Table 4. Our method significantly outperforms baselines in most automatic metrics. Our model obtains a lower PPL, which illustrates our model is better at producing fluency and grammatical text. Higher Dist-1 and Dist-2 scores show our method produces less repetitive unigrams and bigrams during generation, which is essential in creative language generation. The Meta (metaphor) score shows that our model produces more literal expressions than baselines, which might result from the self-training procedure, where non-metaphorical sentences are sometimes wrongly identified by the metaphor identification module, and hence introduces some noise in the metaphor learning process.

We implemented an ablation study to test the effectiveness of self-training and metaphor weighting. Experimental results show the effectiveness of the self-training mechanism in improving both generation fluency and diversity. It can also be observed that removing self-training from our model affects four automatic metrics by a large margin. The metaphor weighting mechanism mainly helps improve the metaphoricity of the generated metaphors and thus improves the Meta score.

5.2 Human Evaluation

We select 180 sentences in total for human evaluation, where the results are shown in Table 4. The Table also shows the inter-annotator agreement of human annotation via Krippendorff’s alpha. We

Methods	Text (Chinese)	Text (Translated)	Con.	Cre.
GPT2	秋天是美丽的，让人赏心悦目。	Autumn is beautiful, and is delightful to the eye.	-	-
	山是翠绿的，像一块无暇的宝石。	Mountain is green just like an innocent jade.	4.0	3.5
	爱情就像是蜂蜜，蜜蜂采蜂蜜。	Love is like honey, bees collect honey.	2.3	1.3
SCOPE	秋天象征春天，像一个月前。	Autumn is a symbol of spring, like a month ago.	-	-
	山是激情的，就像一个爱人。	Mountain is a symbol of passion, like a lover.	1.3	2.3
	爱情是浪漫的，像一盒巧克力。	Love is a symbol of romance, like a box of chocolate.	3.3	2.7
Our method	秋风是那么轻，从远处飘来，就像一条银色的绸带，拂着你，使你感到舒畅，心生向往	The autumn wind is so light, floating from a distance, like a silver ribbon, brushing against you, making you feel comfortable and yearning	5.0	3.7
	山是那神的眼睛，像一盏盏璀璨明灯。	Mountains are the eyes of the god, like bright lights.	3.3	5.0
	恋爱像赌博，我就像赌场的赌徒，赌到手了就赢了，结果却输了。	Love is like gambling, I was the gambler, if I win I will get you, but I lost.	4.3	4.0

Table 5: Example metaphors generated by our method and baselines. **Con.** and **Cre.** indicate the two human evaluation metrics **Consistency** and **Creativity** respectively. We do not assign Con. and Cre. score for non-metaphorical utterances. More examples of our method are shown in Appendix D.

can see that our method beats five baseline models on all three human-centric metrics. The most significant improvement lies in Consistency and Creativity, which shows our method not only can generate creative comparisons, but also can provide a consistent context for each nominal metaphor, which is essential for readability and explainability. Human evaluation also demonstrates the effectiveness of self-training. Self-training enhances generation quality in both fluency and creativity aspects. Metaphor weighting mechanism shows less effectiveness in human evaluation as metaphor weighting mainly aims to improve the metaphoricity of the generated output and human metrics do not measure this aspect. We provide a visualisation for metaphor weighting in Appendix E.

5.3 Case Study

We show some generated examples of GPT2, SCOPE, and our model in Table 5, where the corresponding Consistency and Creativity scores are also provided. Specifically, models generate metaphors by taking different target word as the input. We see that although all three models are able to produce metaphorical outputs, the quality of the generated results differs among systems. **First**, in some cases, baseline model seem fail to generate

metaphorical outputs. For example, when feeding “*autumn*” as input to GPT2 and the SCOPE model, both fail to produce a metaphor but a literal description. **Second**, the comparisons given by our model are more creative than baselines, where GPT2 and SCOPE tend to generate with some common metaphor patterns. For example, GPT2 compares mountain with “**jade**”, which is a very common metaphor in Chinese. **Finally**, we find our method generates metaphors in a relatively more complicated structure and speaks in a more poetic way. For example, our method does not employ a single word in constructing comparison; instead, it tend to generate detailed phrases such as “*love is a gambling and I was the gambler*”, “*autumn wind is like a ribbon brush against you*”. These detailed components paint a more vivid picture, and thus improve the overall readability of the metaphors. The corresponding human-rated Consistency and Creativity scores also support this observation.

6 Conclusion

In this paper, we introduce a novel neural metaphor generator and construct the first Chinese metaphor corpus. The proposed model can effectively learn potential metaphors in unlabelled corpus by self-training and emphasise metaphor-related words in

the metaphor modelling process with metaphorical word identification. Experimental results show that our method is able to generate metaphors of good readability and quality with limited labelled training data. In future work, we plan to generate metaphor with richer linguistic constructs and extend our approach to metaphor generation of other languages.

References

- Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. 2019. Comet: Commonsense transformers for automatic knowledge graph construction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4762–4779.
- Tuhin Chakrabarty, Smaranda Muresan, and Nanyun Peng. 2020. Generating similes effortlessly like a pro: A style transfer approach for simile generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6455–6469.
- Tuhin Chakrabarty, Xurui Zhang, Smaranda Muresan, and Nanyun Peng. 2021. Mermaid: Metaphor generation with symbolism and discriminative decoding. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4250–4261.
- Sam Glucksberg. 1989. Metaphors in conversation: How are they understood? why are they used? *Metaphor and Symbol*, 4(3):125–143.
- Junxian He, Jiatao Gu, Jiajun Shen, and Marc’ Aurelio Ranzato. 2019. Revisiting self-training for neural sequence generation. *arXiv preprint arXiv:1909.13788*.
- Klaus Krippendorff. 2011. Computing krippendorff’s alpha-reliability.
- Saisuresh Krishnakumaran and Xiaojin Zhu. 2007. Hunting elusive metaphors using lexical resources. In *Proceedings of the Workshop on Computational approaches to Figurative Language*, pages 13–20.
- Chee Wee Leong, Beata Beigman Klebanov, Chris Hamill, Egon Stemle, Rutuja Ubale, and Xianyang Chen. 2020. A report on the 2020 *vua* and *toefl* metaphor detection shared task. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 18–29.
- Chee Wee Leong, Beata Beigman Klebanov, and Ekaterina Shutova. 2018. A report on the 2018 *vua* metaphor detection shared task. In *Proceedings of the Workshop on Figurative Language Processing*, pages 56–66.
- Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and William B Dolan. 2016. A diversity-promoting objective function for neural conversation models. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 110–119.
- Lizhen Liu, Xiao Hu, Wei Song, Ruiji Fu, Ting Liu, and Guoping Hu. 2018. Neural multitask learning for simile recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1543–1553.
- Sree Hari Krishnan Parthasarathi and Nikko Strom. 2019. Lessons from building acoustic models with a million hours of speech. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6670–6674. IEEE.
- Anthony M Paul. 1970. Figurative language. *Philosophy & Rhetoric*, pages 225–248.
- Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.
- Yunfan Shao, Zhichao Geng, Yitao Liu, Junqi Dai, Fei Yang, Li Zhe, Hujun Bao, and Xipeng Qiu. 2021. Cpt: A pre-trained unbalanced transformer for both chinese language understanding and generation. *arXiv preprint arXiv:2109.05729*.
- Yosef Ben Shlomo and Mark Last. 2015. Mil: automatic metaphor identification by statistical learning. In *Proceedings of the 2nd International Conference on Interactions between Data Mining and Natural Language Processing-Volume 1410*, pages 19–29.
- Gerard Steen. 2010. *A method for linguistic metaphor identification: From MIP to MIPVU*, volume 14. John Benjamins Publishing.
- Chang Su, Shuman Huang, and Yijiang Chen. 2017. Automatic detection and interpretation of nominal metaphor based on the theory of meaning. *Neurocomputing*, 219:300–311.
- Chang Su, Jia Tian, and Yijiang Chen. 2016. Latent semantic similarity based interpretation of chinese metaphors. *Engineering Applications of Artificial Intelligence*, 48:188–203.
- Asuka Terai and Masanori Nakagawa. 2010. A computational system of metaphor generation with evaluation mechanism. In *International Conference on Artificial Neural Networks*, pages 142–147. Springer.
- Tony Veale. 2016. Round up the usual suspects: Knowledge-based metaphor generation. In *Proceedings of the Fourth Workshop on Metaphor in NLP*, pages 34–41.
- Xijie Wang. 2004. *HanYu XiuCi Xue*. The Commercial Press.

Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. 2020. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10687–10698.

Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. 2017. Seqgan: Sequence generative adversarial nets with policy gradient. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Jiali Zeng, Linfeng Song, Jinsong Su, Jun Xie, Wei Song, and Jiebo Luo. 2020. Neural simile recognition with cyclic multitask learning and local attention. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9515–9522.

Zhengyan Zhang, Xu Han, Hao Zhou, Pei Ke, Yuxian Gu, Deming Ye, Yujia Qin, Yusheng Su, Haozhe Ji, Jian Guan, et al. 2020. Cpm: A large-scale generative chinese pre-trained language model. *arXiv preprint arXiv:2012.00413*.

Jin Zhou. 2020. Love is as complex as math: Metaphor generation system for social chatbot. In *Chinese Lexical Semantics: 20th Workshop, CLSW 2019, Beijing, China, June 28–30, 2019, Revised Selected Papers*, volume 11831, page 337. Springer Nature.

A Chinese Literature Corpus (CLC)

CLC consists of three main categories of Chinese literature: Children’s Literature (Children), Chinese Literature (Chinese), Translated Literature (Translated). Statistics of each category are shown in Table 6.

Category	#Books	#Tokens	#Sentences
Children	195	17M	0.58M
Chinese	336	64M	2.2M
Translated	854	121M	4.2M

Table 6: Summary of CLC.

B SCOPE Model

SCOPE model takes a literal expression as input and produces a simile correspondingly. For example, given “the city is beautiful”, SCOPE model will transfer the literal expression into a simile: “The city is like a painting”.

In our experiments, to compare SCOPE with our method, we first 1) feed a TENOR to COMET (Bosselut et al., 2019) model, to get properties of the TENOR. For example, given a query “<Autumn, SymbolOf>”, COMET predicts a list of properties for Autumn: “Passion, gold” etc. We then 2) construct literal expressions using the TENOR and its

properties. For example, “Autumn is a symbol of passion” is obtained. 3) The literal expression is fed to SCOPE model and a simile is produced. For example, “Autumn is like a lover” is produced by SCOPE model. 4) At last, the simile are concatenate with its literal expression to form a complete NM with context: “Autumn is a symbol of passion, like a lover”.

C Meta Metric

The CMC corpus is splitted into training set (80%) and test set (20%) for training the classifier. We simply add a linear layer plus a binary softmax layer on the RoBERTa model as the NM classifier. The accuracy of the classifier tested on test set of CMC is 97.89%.

D More Examples

Table 7 shows generations produced by our method given different TENORS.

E Visualization of Metaphor Weighting

The visualization of metaphor weighting mechanism is shown in Table 8.

Text (Chinese)	Text (Translated)
爱像一缕金光，即使在黑夜也能照亮你的心灵。	Love is like a ray of golden light, which can illuminate your heart even at night.
爱像一盏明亮的夜灯，让迷途的航船找到港湾；	Love is like a bright night light, let the lost ship find the harbor.
时间像利剑一样无情的锋刃，一旦出鞘，瞬间就割断你人生的纽带。	Time is a ruthless blade like a sharp sword. Once it comes out of the scabbard, it will cut off the bond of your life in an instant.
秋天像个美人的画笔调侃着大地：世界上再没有比这更美的了。	Autumn teases the earth like a beautiful brush: there is nothing more beautiful in the world.
爱心像一片照射在冬日的光，使饥寒交迫的人感到人间的温暖。	Love is like a piece of sunshine in winter, which makes hungry and cold people feel the warmth of the world

Table 7: More generation examples of our method.

Meta Score	Examples	Examples (Translated)
0.984	煤油灯光，仿佛是大海湾里的渔灯野火。	The kerosene lamp is like a fishing lamp wildfire in the bay.
0.893	大公豺皮毛亮得像天边的云霞。	The fur of the cormorant is like the clouds in the sky.
0.887	这声音就像无形的闪电一般。	This sound is like invisible lightning.
0.736	它们很像一对孪生兄弟。	They are much like twin brothers.
0.213	漂泊，会让他见识到他没有见到过的东西。	Wandering will let him see things he has never seen before.

Table 8: Sentences in Chinese literature corpus with its metaphor probability and the visualization of weights in metaphor weighting mechanism are presented for each token.