Nominal Metaphor Generation with Multitask Learning

Yucheng Li [1], Chenghua Lin [2], Frank Guerin [1]
[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@sheffild.ac.uk

Abstract

Nominal metaphors are frequently used in human language and have been shown to be effective in persuading, expressing emotion, and stimulating interest. This paper tackles the problem of Chinese Nominal Metaphor (NM) generation. We introduce a novel multitask framework, which jointly optimizes three tasks: NM identification, NM component identification, and NM generation. The metaphor identification module is able to perform a self-training procedure, which discovers novel metaphors from a large-scale unlabeled corpus for NM generation. The NM component identification module emphasizes components during training and conditions the generation on these NM components for more coherent results. To train the NM identification and component identification modules, we construct an annotated corpus consisting of 6.3k sentences that contain diverse metaphorical patterns. Automatic metrics show that our method can produce diverse metaphors with good readability, where 92% of them are novel metaphorical comparisons. Human evaluation shows our model significantly outperforms baselines on consistency and creativity.

1 Introduction

Metaphors are commonly used in human language. Usually, metaphors compare two different kinds of objects or concepts with the intent to make the expression more vivid, or to make unfamiliar things easier to understand (Paul, 1970). According to contrastive studies of English and Chinese, metaphors are especially crucial in Chinese as there are fewer abstract words in Chinese, so that people tend to express abstract meaning via metaphors (Lian, 1994).

In this paper, we focus on the generation task of a special type of Chinese metaphor – Nominal Metaphors (NMs). NMs (比喻 in Chinese) are figures of speech associating a noun with another noun through a COMPARATOR such as *like, be, become* in English and 像,是,变成 in Chinese. Examples and NM components are shown in Table 1. In addition to the COMPARATOR (**bold**) there are three other components in a nominal metaphor: TENOR, VEHICLE, and CONTEXT (text with underline). The TENOR is the subject of the metaphor, and the VEHICLE is the source of the imagery (i.e., the object of metaphor). CONTEXT is used to explain the comparison and is crucial for understanding the comparison (more details about NM and NM components in § 2.1). The NM generation task is as follows: given a TENOR, generate a metaphor containing the three remaining NM components, i.e., VEHICLE, COMPARATOR and CONTEXT. Previous efforts on NM processing mainly engage in identification (Liu et al., 2018; Zeng et al., 2020) and interpretation (Su et al., 2016, 2017), generation of NMs has not been well studied, despite the benefits it can bring to many downstream tasks. Glucksberg (1989); Zhou (2020) suggest that metaphors are important to an engaging conversation and can effectively stimulate user interest in communicating with chatbots. Chakrabarty et al. (2020, 2021) show that users

1. 这个[孩子]$_{\text{tenor}}$ 壮的像[牛]$_{\text{vehicle}}$ This [boy]$_{\text{tenor}}$ is as strong **as** a [bull]$_{\text{vehicle}}$.	*Nominal*
2. [生活]$_{\text{tenor}}$好比[旅行]$_{\text{vehicle}}$, 没有计划就难以前行 [Life]$_{\text{tenor}}$ **is** a [journey]$_{\text{vehicle}}$, we cannot move on without a plan.	*Nominal*
3. Meta股价[跳水]$_{\text{metaphorical}}$ META stock price [dives]$_{\text{metaphorical}}$.	*Verbal*
4. 他可以像大厨一样烹饪 He can cook like a pro.	*Literal*

Table 1: Examples of Chinese nominal metaphor, verbal metaphor, and NM components. Note that when the words "like" or "as" are used as COMPARATORS, we also call these special NMs 明喻 (Similes).

Proceedings of the 15th International Conference on Natural Language Generation, pages 225 - 235
July 18-22, 2022 ©2022 Association for Computational Linguistics

prefer stories and poems enhanced with metaphor generation by replacing literal expressions with generated metaphors.

To tackle the Chinese NM generation, there are mainly two challenges to address. First, existing Chinese NM corpora are not large enough to power current data-driven text generation approaches. Second, the auto-regressive nature of generative models always assigns higher priority to fluency, which makes the metaphor generation procedure produce *inconsistency errors* (i.e., generating nonsense comparisons without CONTEXT explaining) [1] and *literal errors* (i.e., generating literal expressions).

We propose a novel multitask approach for Chinese NM generation called MetaGen to address the above mentioned problems. Specifically, three tasks are jointly optimized: NM generation, NM identification, and NM components identification. **First**, for the data scarcity problem, we perform a self-training procedure to learn newly discovered metaphors from large-scale unlabeled datasets. Self-training has three main steps: 1) our model is trained on a labeled dataset for NM identification; 2) we apply our model on an unlabeled corpus to detect potential NMs with a corresponding confidence score; and 3) train an NM generation model on the combination of labeled and newly found NMs. By exploiting rich metaphors from large-scale resources, the performance of MetaGen can be significantly improved yet the data requirement can be dramatically reduced. **Second**, MetaGen proposes to identify potential metaphor components (i.e., TENOR, COMPARATOR and VEHICLE) supervised by the attention weights generated by the NM classifier. To alleviate *inconsistency errors*, MetaGen conditions the generation process on the potential NM components; this enforces the CONTEXT generation to depend on the comparison, rather than producing fluent but bland CONTEXT that does not explain the comparison. In terms of the *literal errors*, NMs components are emphasised via attention weight to encourage MetaGen produce metaphorical expressions rather than literals.

We also build an annotated corpus for Chinese NM identification consisting 6.3k sentences. Instead of focusing on a specific metaphorical pattern (Liu et al., 2018), our corpus contains diverse nominal metaphorical usages. We also ensure the CONTEXT is explicit for each metaphor annotated, and the TENOR of each metaphor is also identified. Source code and data can be found in `https://github.com/liyucheng09/Metaphor_Generator`.

2 Related Work

2.1 Metaphors in Chinese

Following (Krishnakumaran and Zhu, 2007; Rai and Chakraverty, 2020), we can divide English metaphors into four types as follows:

Type-I: (Nominal Metaphors) A noun is associated with another noun through the comparators, e.g., “Love is a journey”.

Type-II: (Verbal Metaphors or Subject-Verb-Object (SVO) metaphors) Sentences with metaphorical verb, e.g., “He kills a process”.

Type-III: (Adjective-Noun (AN) metaphor) Metaphorical adjectives with a noun fall into this category, e.g., “sweet boy”.

Type-IV: (Adverb-Verb (AV) metaphor) Metaphorical adverbs with a verb, e.g., “speak fluidly”.

However, the definition of metaphor in the context of Chinese is slightly different from its English counterpart (Wang, 2004). 比喻 (Metaphor), or 打比方 (draw an analogy), which draws a comparison between objects or concepts, mainly means *Type-I* metaphor, i.e., NMs. A specific term 比拟 (Personification/Match) is used to indicate *Type-II, Type-III, Type-IV* metaphors in Chinese, which aims to describe an object or concept in a view of a person or another object. Verbal Metaphors (VMs) are the most frequent type of metaphor in English (Martin, 2006; Steen, 2010), but NMs are the dominant figurative language in Chinese. According to a small scale annotation analysis (Su et al., 2016), NMs are around four times more frequent than VMs in Chinese. Lian (1994) gives a possible explanation for this phenomenon: Chinese people tend to express abstract concepts via nominal metaphors or idioms as there are fewer abstract terms in Chinese than in English. For example, a Chinese nominal metaphor “像竹篮打水” (doing something is like ladling water to a leaky basket), is used to express the meaning of “hopeless”.

Chinese NMs often consist of four components: TENOR, VEHICLE, COMPARATOR, (本体,喻体,比喻词 in Chinese) and CONTEXT, as shown in table 1. The CONTEXT here is a component used

[1] An example of *inconsistency error*: “Teacher is like a candle, floating gently in the air”. Although the comparison is valid, the CONTEXT is inconsistent with the comparison. This also shows the importance of CONTEXT in NM generation.

to *explain* the comparison; its definition is relatively flexible. Sometimes it can be a simple adjective, sometimes a relative clause, or even implicit in some cases. For example, the NM "The city is like a painting" omits the textual CONTEXT to emphasize visual senses. However, CONTEXT is extremely important in helping readers to understand the comparison. According to Indurkhya (2007) and Lakoff and Johnson (2008), a comparison can be drawn between any concepts, but it must have a CONTEXT to explain the comparison or to make the comparison coherent to daily experience. Considering the importance of CONTEXT, we **do not** consider a comparison without CONTEXT as a successfully generated NM case in our experiments. Additionally, there are two linguistic principles Chinese NMs must obey (Wang, 2004): 1) The comparison must be drawn between two concepts with different natures; and 2) the two concepts being compared should share commonalities. Specifically, the COMPARATOR "like" in the example No.4 does not necessarily make it an NM, since the comparison is drawn between the same concept "me cooking" and "pro cooking". The second principle also emphasises the importance of CONTEXT. In summary, even though NMs usually share a relatively simple structure, Chinese NM generation is still challenging due to the requirement of providing CONTEXT and the necessity of understanding the relation between TENOR and VEHICLE.

2.2 Computational Processing of NMs

Previous works on computational processing of NMs can be classified into detection, interpretation and generation.

Detection and Interpretation 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 lack of hyponymy relationship from (Krishnakumaran and Zhu, 2007) and realize 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. Su et al. (2016, 2017) focus on NM interpretation and perform experiments on both English and Chinese NMs. They extract properties of TENOR and VEHICLE from WordNet and use pre-trained word2vec embeddings to identify related properties shared by both components.

Generation Despite the benefits NM generation can bring to the community, prior works on this task are relative sparse. Early works often rely on templates. Terai and Nakagawa (2010) compute the relatedness between concepts with computational language analysis and select candidates to fill metaphor templates like "A is like B". Veale (2016) uses a knowledge-base to generate XYZ style NMs such as "Bruce Wayne is the Donald Trump of Gotham City". Zhou (2020) not only choose candidate concept pairs by word embedding similarity to fill the template but also choose appropriate COMPARATORS to link the concept pair. (Chakrabarty et al., 2020) introduce a neural style transfer approach for simile generation, which finetunes a pre-trained sequence-to-sequence model on a literal-simile parallel dataset. Nevertheless, previous template-based approaches heavily constrain the diversity of generated NMs and both template methods and neural methods produce NMs in a reletive simple structure. Most importantly, previous methods do not provide CONTEXT in their generations (or only provide little CONTEXT), which makes generated results less readable.

3 Method

Given an object or concept as a starting TENOR, a Chinese nominal metaphor will be generated consisting of four NM components: a comparison between TENOR and VEHICLE linked with a COMPARATOR and a CONTEXT as an explanation for the comparison. The overall multitask framework is shown in Figure 1. We can roughly divide our framework into four elements: 1) the GPT2 (Radford et al., 2019) pre-trained language model; and three task-specific fully-connected layers used for 2) NM identification; 3) NM components identification; and 4) NM generation.

3.1 Shared Representation

Since we are tackling a generation task, we employ a pre-trained unidirectional transformer-based language model, GPT2, as our basic encoder. Contextualized words' representations are obtained after feeding words to the GPT2 model. Formally, given sentence $S = (w_0, ..., w_n, w_{EOS})$,

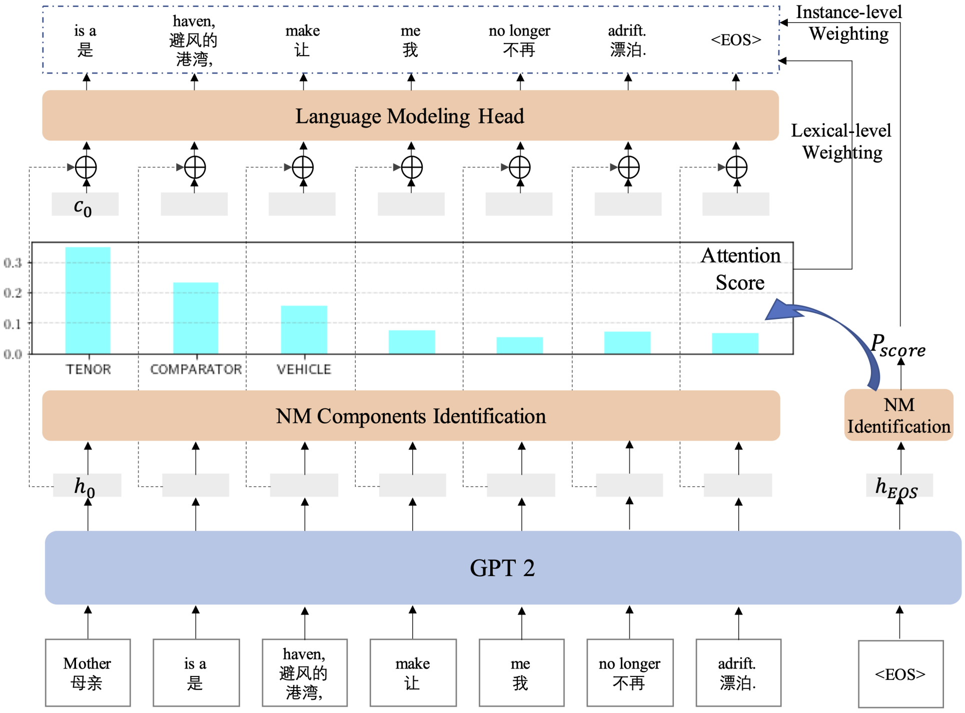

Figure 1: The overall framework.

the GPT2 model produces a list of vectors $H = (h_0, ..., h_n, h_{EOS})$, where the EOS is a special delimiter indicating the end of the sentence. Note that the representation here are used in the three individual tasks described below and the parameters are also shared across all tasks.

3.2 Task 1: NM Identification

The NM identification module is used to assign metaphorical probability to sentences. This score will be used in the Self-Training procedure (described in §3.4). Specifically, we use h_{EOS} as the sentence representation of S (similar to the usage of cls embedding in BERT-based systems (Devlin et al., 2018)) and apply a linear layer plus a softmax layer on it to compute the metaphorical probability of the sentence S. Formally, the metaphor probability is computed as follow:

$$P_M = \text{softmax}(W_m h_{EOS} + b_m) \quad (1)$$

where W_m and b_m is a trainable weight and bias for NMs identification.

We train this module on a supervised dataset noted as $U = \{(x_i, y_i)\}_{i=1}^{N}$, where x indicates the sentence and y indicates whether x is a NM. In summary, we minimize the following loss function for NM identification:

$$L_1 = -\sum_{x \in U} \log P(\hat{y}|x) \quad (2)$$

3.3 Task 2: NMs Components Identification

Although GPT2 model is powerful in generating fluent and grammatical text, it still suffers from incoherence issues (Ko and Li, 2020; Tan et al., 2021). In the scenario of NM generation, it means the CONTEXT generation might be inconsistent with the metaphorical comparison thus resulting in *inconsistency errors*. Besides, the innate tendency of generative models to produce literal text often leads to *literal errors* (Chakrabarty et al., 2021).

To address the *inconsistency errors*, our model conditions the generation procedure on the metaphorical comparison, that is the NM components TENOR, VEHICLE, and COMPARATOR. We also weight these NM components with higher score during training to alleviate *literal errors*. These two approaches (described in §3.4) require the involvement of NM components, therefore, we apply a linear layer to compute the probability for each token to be an NM component. Formally, this probability is computed as follows:

$$P_c = \text{Sigmoid}(W_c H + b_c) \quad (3)$$

where W_c and b_c are trainable weights and bias for NM component identification, and P_c is the resulting probability distribution. Note that this process does not predict the type of components (e.g., TENOR), instead, it only computes a probability for each token indicating the extent to which the generation should focus on each.

We propose to use the attention weights generated from the NM classifier (obtained in §3.2) as the supervision signals for NM component identification. As shown in (Liu et al., 2018; Zeng et al., 2020), the metaphor classifier tends to focus more on corresponding metaphor components, we thus use this property to discover NM components. Specifically, we use KL divergence to have our distribution P_c as close as possible to the attention weights Φ.

$$L_2 = D_{KL}(P_c \| \Phi) \tag{4}$$

where Φ is the self-attention score the h_{EOS} attending to other tokens generated by the last layer Transformer of GPT2.

$$\Phi = \mathrm{softmax}(\frac{Qk^T}{\sqrt{d_k}}) \tag{5}$$

The Q here is the Query matrix for self-attention, and k is the Value vector only for the `EOS` token.

3.4 Task 3: NM Generation

We perform the NM generation task with three steps: 1) conditioning the generation on NM components; 2) emphasizing the NM components; and 3) executing the self-training procedure.

Conditioning To allow token predictions conditioned on NM components, we provide a list of NM component representations $C = (c_0, .., c_i, .., c_n)$ for each prediction step respectively. Then the NM component representation c_i is fed into the language modeling head together with the contextualized token embedding h_i. Formally, c_i is computed as follows:

$$c_i = \sum_{k=0}^{i} \alpha_k \cdot h_k \tag{6}$$

where the weight score α is computed as follows:

$$(\alpha_0, ..., \alpha_i) = \mathrm{softmax}\ P_c^{\{0,...,i\}} \tag{7}$$

The c_i here mainly captures NM component information before the i-th token (i.e., NM components within $(w_0, ..., w_i)$). Then we concatenate the contextualized token embedding h_i and its corresponding NM component information embedding c_i to predict the next token.

$$P(w_{i+1}|w_0, ..., w_i) = \mathrm{softmax}\left[W_l(h_i \oplus c_i) + b_l\right] \tag{8}$$

where the W_l and b_l are trainable weight matrix and bias, $\oplus$ indicating the concatenation operation.

Emphasizing We emphasize the NM components during training by directly applying attention weight P_c on the loss function. Specifically, given a sentence $S = (w_0, ..., w_n)$, we minimize the following loss function:

$$\mathcal{L}(S) = -\sum_{i=0}^{n} P_c^i \cdot \log P(w_i|w_0, ..., w_{i-1}) \tag{9}$$

where P_c^i is the probability to be one of the NM components of token w_i.

Self-training Self-training is an effective approach to tackle data scarcity and has been successfully used in many downstream tasks (He et al., 2019; Parthasarathi and Strom, 2019; Xie et al., 2020). In our setting, we adopt self-training for discovering novel Chinese NMs from large-scale corpora to train the NM generation module so that the fluency and diversity of generation can be improved.

Formally, given an unlabeled corpus $V = \{x_i\}_{i=0}^N$ where each x is a sentence $x = (w_0, ..., w_n)$, the NM identifier will assign a probability to each x_i noted as P_M^i. We than mix the unlabeled corpus $V = \{(x_i, P_M^i)\}_{i=0}^N$ and supervised dataset $U = \{(x_i, y_i)\}_{i=1}^N$ together, and train the overall framework on it. Formally, we minimize the following loss function:

$$L_3 = -\left[\sum_{x \in V} P_M^i \cdot \mathcal{L}(x) + \sum_{S \in U} \mathcal{L}(S)\right] \tag{10}$$

3.5 Training and Inference

The final loss function of our framework is a weighted sum of three task-specific loss function.

$$L = \gamma \cdot L_1 + L_2 + L_3 \tag{11}$$

Note that when learning unlabeled sentences, γ is set to 0, since these instances lack the supervision label for NM identification. To help our model converge, before training the overall framework on the mixed data by L, we pre-train our model on the supervised dataset for Task 1 first. Besides, when doing inference, our model only performs Task 3.

4 Experiment

4.1 dataset

To train our multitask framework, we construct two datasets: a supervised Chinese NM Corpus (CMC)

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

Table 2: Statistics of CMC and CLC datasets

and a large-scale unsupervised Chinese Literature Corpus (CLC).

CMC Existing Chinese metaphor corpus are neither too small, like Su et al. (2016) contains 120 examples, or focusing on a specific metaphorical pattern, like Liu et al. (2018) contains sentences with a specific COMPARATOR 像 (like). In our corpus, we try to include nominal metaphors as diverse as possible. The annotation of the CMC consists of four steps: 1) we collect 55,000 Chinese sentences from essays, articles, and novels; 2) we employ three Chinese graduate students with background of NLP to label each sentence as a NM or not; 3) we take the majority agreement as the final label for each sentence; 4) the boundary of TENOR is identified at last. To encourage the CONTEXT to be generated, we ensure CONTEXT occurs explicitly in each metaphor we labeled. Before the annotation, annotators are trained with examples and instructed with basic Chinese NM principles (described in §2.1). We compute the inner-annotator agreement of NM label via Krippendorff's alpha (Krishnakumaran and Zhu, 2007). The agreement rate is 0.84. Statistics of CMC are shown in Table 2. Some examples are shown in Appendix A.1.

CLC In self-training, we need a large-scale corpus so that the NM identifier can discover novel NMs. However, popular Chinese corpora, such as news, Wikipedia, web pages, are not suited 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.2). Statistics of CLC are shown in Table 2.

4.2 Baselines

Chinese NMs generation is a novel task, we select three general generative models and an English simile generation method as baselines.

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 Chinse NM.

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

BART: We fine-tune a Chinese version BART model (Shao et al., 2021) model on parallel data pairs <TENOR, Sentence> obtained from CMC.

SCOPE: (Chakrabarty et al., 2020) 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 model[2] to avoid starting training from scratch. Our model is pre-trained on NM identification task with CMC for 3 epochs before jointly optimizing three task-specific loss functions. The implementation of SeqGAN[3] and the pre-trained Chinese BART model[4] can be found in the footnote. Before the SeqGAN starts training on CMC, we first pre-train the generator of SeqGAN on CLC for 50k steps. Hyper-parameters not specified are all followed by 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 input TENORS into English (via Google Translator), then translate generated NMs back to Chinese (details in Appendix B). In the test stage, we randomly select and feed 200 TENORS from CMC to all generative models. During decoding, all beam sizes are set to 12, thus each model generated 12 sentence for each TENOR. 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,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

[2] https://huggingface.co/uer/gpt2-chinese-cluecorpussmall

[3] https://github.com/LantaoYu/SeqGAN

[4] https://huggingface.co/fnlp/bart-base-chinese

Methods	PPL	Dist-1	Dist-2	Meta	Novelty	Fluency	Consistency	Creativity
SeqGAN	89.43	.00336	.0116	**.998**	.200	3.33 (.51)	3.80 (.46)	1.67 (.34)
GPT2	57.88	.00916	.1154	.981	.800	4.00 (.62)	3.10 (.39)	2.60 (.31)
BART	48.58	.00826	.0971	.978	.725	4.35 (.54)	3.05 (.37)	2.30 (.32)
SCOPE	92.32	.00517	.0673	.910	.385	3.10 (.64)	2.70 (.44)	2.10 (.45)
Our Method	**25.79**	**.01153**	**.1674**	.948	**.920**	**4.65** (.58)	**4.40** (.45)	**3.80** (.36)
w/o Self-training	62.54	.00674	.0906	.982	.785	3.85 (.54)	3.87 (.42)	2.76 (.38)
w/o Emphasizing	25.58	.01150	.1529	.803	.900	4.50 (.63)	3.91 (.32)	3.41 (.43)
w/o Conditioning	24.93	.01053	.1534	.875	.930	4.25 (.61)	3.05 (.45)	3.24 (.39)

Table 3: 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.

train a RoBERTa-based Chinese NM classifier on CMC to compute the ratio of metaphorical utterances in the generated sentences. The accuracy of this classifier is 97.89%, which is reasonable enough to perform evaluation (details in Appendix C). **Novelty** is to test how well models can generate metaphors they have never seen during training. We use a syntax-based approach to identify TENORS and VEHICLES from generated NMs and compute the proportion of <TENOR, VEHICLE> pair that does not co-occurr in the training set.

Human Evaluation Due to the creative and delicate usage of NM, 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 can explain itself; how well the VEHICLE relate to TENOR and how well the CONTEXT explain the comparison. 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 3. 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 NM identification module, and thus there is noise in NM modeling. The highest Novelty score demonstrates our method's ability to generate creative comparisons.

We implemented an ablation study to test the effectiveness of self-training, NM component emphasizing, and context conditioning. Experimental results prove the self-training mechanism improves both generation fluency and diversity. Removing self-training from our model affects four automatic metrics by a large margin. The NM component emphasizing mainly helps our method alleviate *literal errors* and thus improve the Meta score. The context conditioning also benefits the overall framework in Meta score.

5.2 Human Evaluation

We select 180 sentences in total to annotate (15 TENORS, 12 sentences for each TENOR). Human evaluation results are shown in Table 3. The Table also shows the inter-annotator agreement of human annotation via Krippendorff's alpha. We can see that our method beats four baseline models on all three human-centric metrics. The most significant improvement lies in Consistency and Creativity, which show our method can not only generate creative comparisons, but, most importantly, also provide a CONTEXT for each NM to explain the comparison, which is essential for readability. Human evaluation also demonstrates the effectiveness of self-training, emphasizing, and conditioning. Self-training enhances generation quality in both fluency and creativity dimensions. Conditioning mostly contributes to the consistency score.

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.	-	-
	秋天是个动情的音符，荡漾在夏日的清纯中。	Autumn is an emotional note, rippling in the purity of summer.	2.0	3.5
	秋天是最好的伴奏曲，让世界充满微笑。	Autumn is the best concertos, making the world full of smiles.	3.3	2.0
SCOPE	秋天象征春天，像一个月前。	Autumn is a symbol of spring, like a month ago.	-	-
	秋天象征热情，像一个情人。	Autumn is a symbol of passion, like a lover.	3.3	1.3
	秋天象征爱情，像一个女人。	Autumn is a symbol of love, like a woman.	1.3	2.7
Our method	秋天像一只彩笔画般的画笔，勾勒出一幅幅多彩多姿的画卷。	Autumn is like a multi-colored paintbrush, sketching out colorful pictures.	5.0	2.7
	秋天像小姑娘的脚，带着她那柔软的臂膀，在枝头翩翩起舞。	Autumn is like a little girl's feet with her softness. Arms, dancing in the branches.	3.7	5.0
	秋天像刚刚落地的苹果,在果园里露出个头。	Autumn is like an apple that has just fallen, showing its head in the orchard.	4.3	4.0
	秋天像刚落的蝉，婉转地鸣叫着，见证着树梢上金黄色的叶子慢慢向蓝天生长。	Autumn is like a cicada that has just fallen, chirping tactfully, seeing the golden leaves on the treetops grow towards the blue sky slowly.	4.3	5.0

Table 4: NMs generated by our method and baselines given a TENOR 秋天 (Autumn). **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 MetaGen are shown in Appendix D.

5.3 Case Study

Generated examples of GPT2, SCOPE, and our model are shown in Table 4. The corresponding Consistency and Creativity score are also given. In this table, models generate NMs by taking 秋天 (Autumn) as the input TENOR. We see that although all three models are able to produce metaphorical outputs, the quality of generated results differs among systems. **First**, the comparisons given by our model are more diverse than baselines. We can identify similar patterns in the outputs of GPT2 and SCOPE. For example, GPT2 tends to compare autumn with "music" (i.e., note and accompaniment) and SCOPE is likely to relate autumn with love (i.e., lover and woman). **Second**, CONTEXT produced by our method can explain the comparison well, which ensures the consistency and readability of the outputs. However, baselines are either give little CONTEXT (like SCOPE gives an adjective or noun as CONTEXT) or inappropriate CONTEXT (like GPT2 uses summer in the comparison of autumn). **Third**, we find our method generates NMs in a relatively more complicated structure and speaks in a more poetic way. For example, our method does not use a single word as VEHICLE, instead, it generates detailed phrases, such as "apple that has just fallen", "dancing on the branches". These detailed components paint a more vivid picture, and thus improve the overall readability. The corresponding human-rated Consistency and Creativity scores support this.

6 Conclusion

In this paper, we introduce a novel language generation task: Chinese nominal metaphor generation. We also propose a multitask framework for Chinese nominal metaphor generation. Additionally, we publish an annotated corpus for Chinese nominal metaphors. Future directions can be trying the usage of syntactic features and controllable NM generation. Moreover, we would also like to evaluate the effect of metaphor generation in downstream tasks, such as story generation, dialog systems, and educational scenarios.

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.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

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. In *International Conference on Learning Representations*.

Bipin Indurkhya. 2007. Creativity in interpreting poetic metaphors. In T. Kusumi, editor, *New directions in metaphor research*, pages 483–501. Tokyo, Japan: Hitsuji Shobo.

Wei-Jen Ko and Junyi Jessy Li. 2020. Assessing discourse relations in language generation from gpt-2. In *Proceedings of the 13th International Conference on Natural Language Generation*, pages 52–59.

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.

George Lakoff and Mark Johnson. 2008. *Metaphors we live by*. University of Chicago press.

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.

Shuneng Lian. 1994. *Contrastive Studies OF English and Chinese*. Fudan University Press.

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.

James H Martin. 2006. A corpus-based analysis of context effects on metaphor comprehension.

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.

Sunny Rai and Shampa Chakraverty. 2020. A survey on computational metaphor processing. *ACM Computing Surveys (CSUR)*, 53(2):1–37.

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.

Bowen Tan, Zichao Yang, Maruan Al-Shedivat, Eric Xing, and Zhiting Hu. 2021. Progressive generation of long text with pretrained language models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4313–4324.

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 Dataset

A.1 Chinese NM Corpus (CMC)

Examples in CMC are shown in Table 5.

A.2 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.

Label	Examples
NM	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.
NM	食堂开饭时，全校同学像热锅上的蚂蚁一样挤成一团。 When the dining hall opened, the whole school huddled together like ants on a hot pot.
Not NM	泛着银光的大海在他身后铺展开来。 The silver-filled sea spread out behind him.

Table 5: Examples of metaphor and not metaphor in the CMC.

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 splited 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.

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 MetaGen.