# Metaphor Detection via Linguistics Enhanced Siamese Network

**Shenglong Zhang** **Ying Liu**
Tsinghua University, Beijing, China, 100084
zsl18@mails.tsinghua.edu.cn
yingliu@mail.tsinghua.edu.cn

## Abstract

In this paper we present MisNet, a novel model for word level metaphor detection. MisNet converts two linguistic rules, i.e., Metaphor Identification Procedure (MIP) and Selectional Preference Violation (SPV) into semantic matching tasks. MIP module computes the similarity between the contextual meaning and the basic meaning of a target word. SPV module perceives the incongruity between target words and their contexts. To better represent basic meanings, MisNet utilizes dictionary resources. Empirical results indicate that MisNet achieves competitive performance on several datasets.

## 1 Introduction

Metaphor is an omnipresent figurative language in daily communication. Conceptual Metaphor Theory proposes that metaphor is a mapping mechanism between the source domain and the target domain (Lakoff and Johnson, 2008).

e.g. 1 *The scream* ***pierced*** *the night.*

In e.g. 1, the literal meaning of the verb ***pierce*** is "some sharp object goes into or on through something" . However, the contextual meaning is "break silence". Here, the source domain is a highly abstract action, while the target domain can present the corresponding meaning in a more concrete way. In general, there exists two types of metaphors, i.e., novel metaphors and conventional metaphors.

e.g. 2 *He* ***attacked*** *the government's defence policy.*

Semantic Shift Theory shows that new lexical senses can derive from metaphors (Blank, 2013). Once a metaphorical usage of a word is accepted by most people, the metaphorical lexical sense is fixed. Thus a polysemant may have metaphorical marginal meanings (Bloomfield, 1994). In e.g. 2, the metaphorical target word ***attack*** means ***criticize***, which is also a sense of it. It is a conventional metaphor for the metaphorical meaning has been fixed. While e.g. 1 is a novel metaphor, for the metaphorical usage is temporary.

Linguistic rules instruct us how to identify metaphors. According to Metaphor Identification Procedure (MIP) (Crisp et al., 2007; Steen, 2010), a metaphor is identified if the contextual meaning of the target word contrasts with one of its more basic meaning. More basic meanings are: **1**) More concrete; what they evoke is easier to imagine, see, hear, feel, smell, and taste; **2**) related to bodily action; **3**) more precise (as opposed to vague); **4**) historically older(Group, 2007; Do Dinh et al., 2018). The basic meaning of ***pierce*** in e.g. 1 contrasts with its contextual meaning so it is a metaphor.

Researchers are divided on how to represent basic meanings. Gao et al. (2018) and Mao et al. (2019) used dynamic ELMo embeddings and static GloVe embeddings to encode contextual target meanings and basic target meanings respectively. Choi et al. (2021) proposed that a target used alone is literal. Su et al. (2021) and Wan et al. (2021) used the gloss (brief definition) of a target to represent its literal meaning. All the methods are not linguistically intuitive, because we do not know whether the basic meaning is accurately encoded through embedding and a basic meaning is not simply the average of its gloss. Since basic meanings are not properly represented, MIP may be invalid. Consequently, conventional metaphors are ignored by previous studies as well(Tong et al., 2021).

Another linguistic rule is Selectional Preference Violation (SPV) (Wilks, 1975, 1978), which suggests that a metaphor is identified by noticing semantic incongruity between a target word and its context. In e.g. 1, ***pierce*** rarely occurs in the context consisting of ***scream*** and ***night***. There comes a contextual contrast in such a collocation. Notice that SPV becomes invalid when faced with a conventional metaphor, because the context is also usual for the metaphor word like ***attack*** in e.g. 2.

To better use linguistic rules, we propose to use

*Proceedings of the 29th International Conference on Computational Linguistics*, pages 4149–4159
October 12–17, 2022.

the sentence where the target adopts its basic meaning for a better representation. This idea is in line with mainstream language models: you shall know a word by the company it keeps (Firth, 1957).

In this paper, we propose a novel metaphor detection model named **M**etaphor **I**dentification from **S**iamese **Net**work (MisNet). MisNet adopts a siamese framework, consisting of two separate encoders . We regard MIP as a representation based semantic matching task between target in the given sentence and target in the basic usage. MIP is accomplished across two encoders. We model SPV as an interaction based semantic matching task between the target word and its context to measure semantic incongruity. SPV is implemented within a single encoder. Based on the fusion of MIP and SPV modules, MisNet makes a final prediction.

The contributions of this paper can be summarized as follows:

- We model two linguistic rules, i.e., MIP and SPV as two semantic matching tasks. Our model is linguistically intuitive and also extensible.
- We use basic usage to better encode the basic meaning for a target word, thus our model can avoid the invalidation of MIP and SPV. It is also proficient in tackling conventional metaphors.
- Experimental results show that our method achieves competitive performance on several datasets over existing approaches.
- Our code is available on GitHub[1].

## 2 Related Work

Recently, metaphor detection has attracted lots of attention. With the development of NLP technologies, various methods have been applied. In general, these approaches can be categorized into three types: feature engineering based, RNN based and transformer based methods.

Feature engineering based methods use linguistic features such as word concreteness, word abstractness (Turney et al., 2011), and word class etc., as input of a certain machine learning model like Logistic Regression and SVM (Shutova and Sun, 2013; Assaf et al., 2013; Tsvetkov et al., 2014; Wan et al., 2020). RNN based models use RNN as a feature extractor to form contextual representations to identify metaphors(Wu et al., 2018; Gao et al., 2018; Mao et al., 2019; Le et al., 2020). Though great improvements have been made, RNN based models mostly use static word representations like Word2Vec and GloVe, so they are not adept at metaphors which convey complex contextual senses. Also, due to the nature of RNN, these models cannot be paralleled.

Transformer based methods use a Pretrained Language Model (PLM) like BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019), as the backbone of the model, yielding promising results in metaphor detection. Su et al. (2020) converted metaphor detection into a machine reading comprehension task with various linguistic features incorporated, achieving best reported results in ACL 2020 metaphor detection shared task. Choi et al. (2021) used a late-interaction mechanism to encode the contextual meaning and the literal meaning of a target word. Song et al. (2021) focused on verb metaphor detection. They used dependency parsing to extract the objects and subjects of the given verbs to use syntactic relations. Lin et al. (2021) utilized contrastive learning to distinguish metaphors from literal usages. They also used self-training strategy to generate pseudo-labels, which largely expanded existing public datasets. Also, some recent researches noticed that external dictionary resources could greatly help metaphor detection. Wan et al. (2021) and Su et al. (2021) used glosses to interpret target words. They took the average embeddings of a gloss as the corresponding target representation. However, the meaning of a word is not simply the average of its definition. There still leaves much space to better represent literal meanings.

## 3 Proposed Model

Some researchers use sequence labeling to detect metaphors, i.e., label all $n$ words in a sentence in one go (Gao et al., 2018; Mao et al., 2019). In this paper, we use word classification paradigm to detect metaphors. We regard each word in the given sentence as target in order, then take a target along with its given sentence to predict the metaphoricity of the target for $n$ times.

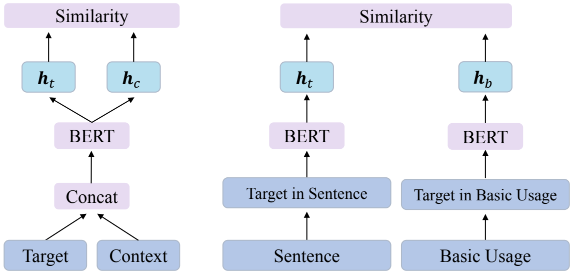
### 3.1 SPV & MIP : Semantic Matching

Semantic matching intends to measure the similarity between two given texts, of which Interaction-

[1] https://github.com/SilasTHU/MisNet

based Models and Representation-based Models are two main paradigms. In this paper, we use the mentioned two semantic matching models to implement SPV and MIP.

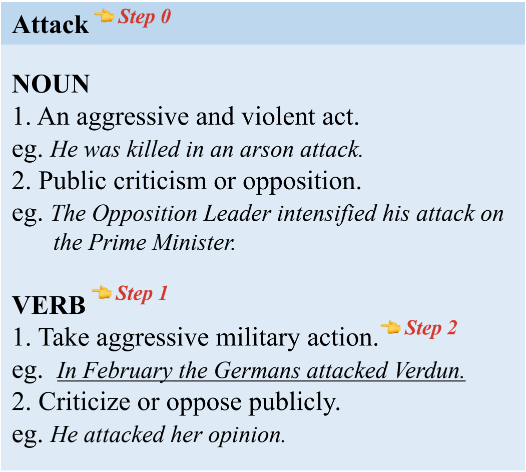
**Interaction-based Model for SPV:** for an Interaction-based Model (IM), two texts are concatenated as input, where each token within the input can fully interact with the others (Yang et al., 2019; Rao et al., 2019). Vanilla BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) take two texts as model input to compute a similarity score, so they are also IMs. SPV suggests to notice the incongruity between the target word and its context, which can be measured through the semantic similarity between them. As Fig. 1 (a) shows, we adopt an IM to implement SPV, because the target and its context are from a same sequence, such that they are naturally concatenated. In our model, they are two texts to be matched. Hence they can interact thoroughly through multi-head self-attention (Vaswani et al., 2017) in BERT. Finally, we manage to retrieve the contextual target embedding $\boldsymbol{h}_t$ and context embedding $\boldsymbol{h}_c$ to calculate the similarity.



(a) Interaction-based Model (b) Representation-based Model

Figure 1: Interaction-based Model and Representation-based Model in semantic matching task.

**Representation-based Model for MIP:** for a Representation-based Model (RM), two texts are input into different encoders to get their representations, so the two texts do not interact with each other during encoding (Conneau et al., 2017; Reimers and Gurevych, 2019). MIP prompts us to determine whether the target has a more basic usage. Thus we calculate the semantic similarity between target in the given sentence and that in the basic usage. As Fig. 1 (b) shows, we model MIP as an RM because using two separate encoders for the given sentence and the basic usage can avoid unnecessary interactions. Hence the contextual meaning and the basic meaning of a target can be better captured. We then obtain the contextual target embedding $\boldsymbol{h}_t$ and the basic embedding $\boldsymbol{h}_b$, based on which the similarity is computed.


**Attack** *Step 0*

**NOUN**
1. An aggressive and violent act.
eg. *He was killed in an arson attack.*
2. Public criticism or opposition.
eg. *The Opposition Leader intensified his attack on the Prime Minister.*

**VERB** *Step 1*
1. Take aggressive military action. *Step 2*
eg. *In February the Germans attacked Verdun.*
2. Criticize or oppose publicly.
eg. *He attacked her opinion.*

Figure 2: Basic Usage Retrieving Strategy for *attack (verb)*. **Step 0.** Find the term of the target word. **Step 1.** Locate at the same POS tag. **Step 2.** Take the example sentence under the first gloss as a basic usage since dictionary editors tend to place more basic meanings in the front.

## 3.2 MisNet Architecture

**Combine MIP and SPV:** Using SPV to detect a metaphor depends on the incongruity of the target and its context. However, a conventional metaphorical target does not own a paradoxical context (the context is usually common for the target), so SPV may be invalid (Haagsma and Bjerva, 2016; Do Dinh et al., 2018). MIP is utilized based on the basic meaning and the contextualized meaning of a target. Since we use basic usages to encode basic meanings, our MIP module is suitable for both conventional and novel metaphors. We leverage the combination of MIP and SPV for better metaphor detection, which is proved to be a better method by experimental results.

Fig. 3 shows the architecture of MisNet. MisNet adopts a siamese framework to combine MIP and SPV. The left part encodes the given sentence, while the right uses the target, the POS tag, and the basic usage. MIP is implemented across the left and the right encoders, while SPV functions within the left one.

**Left Encoder Input**: the left encoder input is the given sentence in the dataset:

$$\mathbf{L} = ([\text{CLS}], \text{given\_sentence}, [\text{SEP}])\,, \qquad (1)$$

where [CLS] and [SEP] are the two special tokens of BERT.

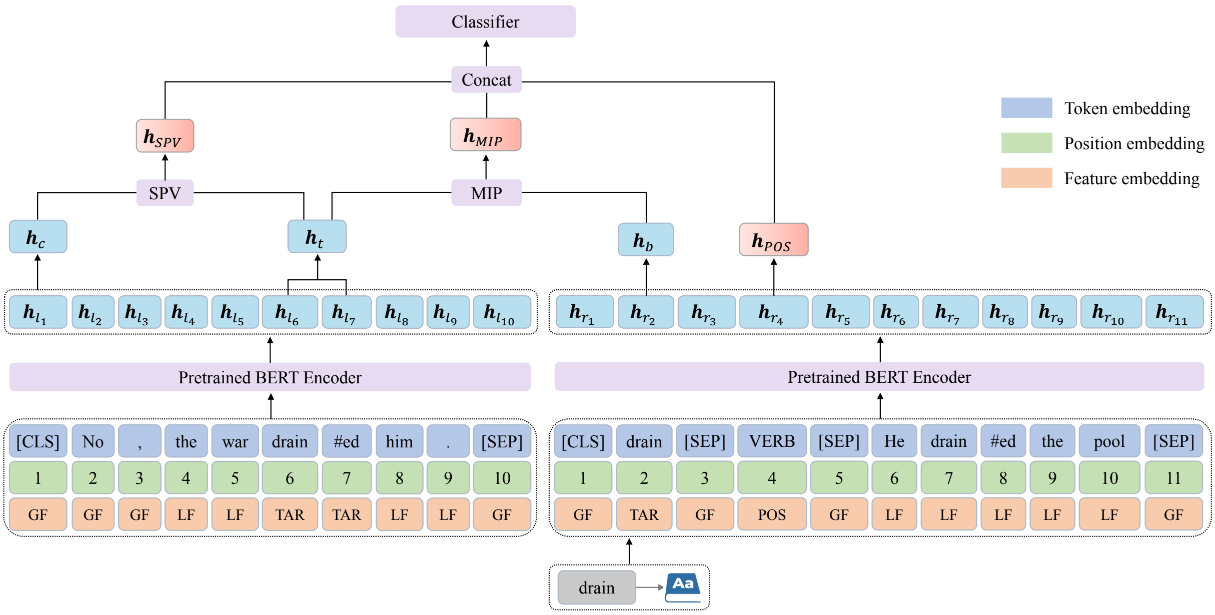

Classifier
Concat
$h_{SPV}$
$h_{MIP}$
SPV
MIP
$h_c$
$h_t$
$h_b$
$h_{POS}$
Token embedding
Position embedding
Feature embedding
$h_{l_1}$ $h_{l_2}$ $h_{l_3}$ $h_{l_4}$ $h_{l_5}$ $h_{l_6}$ $h_{l_7}$ $h_{l_8}$ $h_{l_9}$ $h_{l_{10}}$
$h_{r_1}$ $h_{r_2}$ $h_{r_3}$ $h_{r_4}$ $h_{r_5}$ $h_{r_6}$ $h_{r_7}$ $h_{r_8}$ $h_{r_9}$ $h_{r_{10}}$ $h_{r_{11}}$
Pretrained BERT Encoder
Pretrained BERT Encoder
[CLS] No , the war drain #ed him . [SEP]
1 2 3 4 5 6 7 8 9 10
GF GF GF LF LF TAR TAR LF LF GF
[CLS] drain [SEP] VERB [SEP] He drain #ed the pool [SEP]
1 2 3 4 5 6 7 8 9 10 11
GF TAR GF POS GF LF LF LF LF LF GF
drain
Aa

Figure 3: MisNet architecture. The two BERT encoders share weights. $\boldsymbol{h}_c$, $\boldsymbol{h}_t$, $\boldsymbol{h}_b$ are context embedding, contextual target meaning, and basic meaning respectively. GF, LF, POS, TAR denote global feature, local feature, POS feature and target word.

**Right Encoder Input**: we concatenate the target word, target POS tag and the basic usage as the right encoder input. The basic direct usage (or a more basic usage, which is still in line with MIP rule.) for the target word is retrieved via Basic Usage Retrieving Strategy as Fig. 2 shows. The right input is:

$$\mathbf{R} = ([\text{CLS}], \text{target\_word}, [\text{SEP}], \text{POS}, [\text{SEP}], \text{basic\_usage}, [\text{SEP}]). \tag{2}$$

If we fail to retrieve the basic usage, we just use the target word and its POS tag.

Different parts of the input have different impacts on metaphor detection. Self-Attention mechanism in BERT can benefit semantic representations for input tokens(Vaswani et al., 2017), but it may not be sufficient to notice the differences among various input parts. To treat them differently, for both the left and the right input, we add input type feature embeddings to the BERT input layer. We design four features and embed them into fixed-length vectors:

- **POS Feature**: the POS tag of the target word. It only exists in the right input.
- **Target Feature**: the target word. The left and the right input have a same target word.
- **Local Feature**: following Su et al. (2020) and Choi et al. (2021), we set the clause where the target word lies as local context. For simplicity, a clause is separated by commas, dots, exclamation marks, and question marks etc. Since a basic usage is usually short[2], we regard the whole basic usage as local feature.
- **Global Feature**: other tokens except the POS tag, the target word, and its local feature.

After tokenization via Byte-Pair Encoding (BPE) algorithm (Radford et al., 2019), $\mathbf{L}$ is cut into $n$ tokens, while $\mathbf{R}$ has $m$ tokens. The final input for BERT is token embeddings, positional embeddings, plus feature embeddings. Then we use BERT to get contextualized representations:

$$\mathbf{H}_L = \text{BERT}(\mathbf{L}) = (\boldsymbol{h}_{l_1}, \boldsymbol{h}_{l_2}, \cdots, \boldsymbol{h}_{l_n}), \tag{3}$$
$$\mathbf{H}_R = \text{BERT}(\mathbf{R}) = (\boldsymbol{h}_{r_1}, \boldsymbol{h}_{r_2}, \cdots, \boldsymbol{h}_{r_m}), \tag{4}$$

where $\mathbf{H}_L \in \mathbb{R}^{n \times d}$ and $\mathbf{H}_R \in \mathbb{R}^{m \times d}$ are the embedding matrices of $\mathbf{L}$ and $\mathbf{R}$ respectively. $d$ is the hidden dimension in BERT.

Based on $\mathbf{H}_L$, we can get the contextual meaning of the target word, which is denoted by $\boldsymbol{h}_t$. If the target word is cut into $k$ tokens by BPE, we just take the average:

$$\boldsymbol{h}_t = \frac{1}{k} \sum_{i=u}^{u+k-1} \boldsymbol{h}_{l_i}, \tag{5}$$

[2] It is difficult to get the exact position for a target word in its basic usage, because it may not be rendered in the original form and lemmatization is not always accurate.

where $u$ is the start location for target in the left input. Similarly, based on $\mathbf{H}_R$, we get the basic meaning of the target word, which is denoted by $\boldsymbol{h}_b$. Notice that we do not need to know the exact position of the target word in the basic usage, because transformer encoder will apply self-attention mechanism to make the target word in $\mathbf{R}$ focus on the relevant parts automatically (Vaswani et al., 2017).

For the left input, we take the average of the embedding matrix $\mathbf{H}_L$ to get the context embedding:

$$\boldsymbol{h}_c = \mathrm{Mean}(\mathbf{H}_L). \tag{6}$$

MIP layer compares the basic meaning vector $\boldsymbol{h}_b$ and the contextual target meaning vector $\boldsymbol{h}_t$. We use a linear transformation to implement MIP:

$$\boldsymbol{h}_{\text{MIP}} = W_{\text{MIP}}^{\top}[\boldsymbol{h}_t; \boldsymbol{h}_b; |\boldsymbol{h}_t - \boldsymbol{h}_b|; \boldsymbol{h}_t * \boldsymbol{h}_b] + b_{\text{MIP}}, \tag{7}$$

where $[\cdot]$ is a readout method. $|\cdot|$ means absolute value. ; is concatenation, and $*$ denotes hadamard product. We combine these methods to readout different representations. $W_{\text{MIP}}$ and $b_{\text{MIP}}$ are weight and bias of MIP layer respectively. Similarly, we conduct SPV on context vector $\boldsymbol{h}_c$ and contextual target meaning vector $\boldsymbol{h}_t$:

$$\boldsymbol{h}_{\text{SPV}} = W_{\text{SPV}}^{\top}[\boldsymbol{h}_c; \boldsymbol{h}_t; |\boldsymbol{h}_c - \boldsymbol{h}_t|; \boldsymbol{h}_c * \boldsymbol{h}_t] + b_{\text{SPV}}, \tag{8}$$

where $W_{\text{SPV}}$ and $b_{\text{SPV}}$ are weight and bias of SPV layer respectively. POS information plays an important role in metaphor detection, so we extract POS vector $\boldsymbol{h}_{\text{POS}}$ from the right encoder. Finally, we combine MIP, SPV, and POS information to decide whether the target word is metaphorical:

$$\boldsymbol{y} = \sigma\left(W^{\top}[\boldsymbol{h}_{\text{MIP}}; \boldsymbol{h}_{\text{SPV}}; \boldsymbol{h}_{\text{POS}}] + b\right), \tag{9}$$

where $W$ and $b$ are weight and bias. $\sigma$ is a softmax function. $\boldsymbol{y} \in \mathbb{R}^2$ indicates the predicted label distribution.

### 3.3 Training Objective

For a classification task, we use cross entropy loss as our optimization criterion:

$$\mathcal{L} = -\frac{1}{N}\sum\nolimits_{i=1}^{N} w_{y_i} y_i \log(\hat{y}_i), \tag{10}$$

where $N$ is the count of training samples. $y_i$ and $\hat{y}_i$ denote the ground truth label and the predicted score for the i-th sample respectively. $w_{y_i}$ is class weight to alleviate data unbalance problem.

## 4 Experiments

### 4.1 Datasets

Following most metaphor identification works, we use four widely-used public datasets. The statistic information is shown in Table 1.

**VUA All** (Steen, 2010): The largest metaphor dataset drawn from VU Amsterdam Metaphor Corpus (VUA). VUA collects sentences from the BNC-Baby Corpus, including four genres: academic, conversation, fiction, and news. VUA All dataset labels each word in each POS for each sentence.

**VUA Verb** (Steen, 2010): VUA Verb is a subset of VUA All. VUA Verb dataset only has verb targets.

**MOH-X** (Mohammad et al., 2016): MOH-X dataset focuses on the verb track. MOH-X collects metaphorical and literal usages for verbs from WordNet. Each verb in MOH-X has multiple senses, of which at least one is metaphorical.

**TroFi** (Birke and Sarkar, 2006, 2007): TroFi dataset only includes verb targets. The literal and metaphorical usages for 50 English verbs are drawn from The 1987-89 Wall Street Journal Corpus.

| Dataset | #Sent. | #Target | %Met. | Avg. Len |
|---|---|---|---|---|
| VUA All$_{tr}$ | 6,323 | 116,622 | 11.19 | 18.4 |
| VUA All$_{val}$ | 1,550 | 38,628 | 11.62 | 24.9 |
| VUA All$_{te}$ | 2,694 | 50,175 | 12.44 | 18.6 |
| VUA Verb$_{tr}$ | 7,479 | 15,516 | 27.90 | 20.2 |
| VUA Verb$_{val}$ | 1,541 | 1,724 | 26.91 | 25.0 |
| VUA Verb$_{te}$ | 2,694 | 5,873 | 29.98 | 18.6 |
| MOH-X | 647 | 647 | 48.69 | 8.0 |
| TroFi | 3,737 | 3,737 | 43.54 | 28.3 |

Table 1: Datasets information. **#Sent.**: Number of sentences. **#Target**: Number of target words. **%Met.**: Percentage of metaphors. **Avg. Len**: Average sentence length.

### 4.2 Baselines

**RNN_ELMo** (Gao et al., 2018) and **RNN_BERT** (Mao et al., 2019): two RNN based sequence labeling models. They concatenate embeddings of ELMo (or BERT) and GloVe to represent a word.

**RNN_HG** and **RNN_MHCA** (Mao et al., 2019): RNN_HG uses MIP to compare literal target meanings and contextual target meanings, which are represented by GloVe and ELMo embeddings respectively. RNN_MHCA is based on SPV, with multi-head contextual attention utilized.

**MUL_GCN** (Le et al., 2020): MUL_GCN adopts a multi-task learning framework to tackle metaphor

| Model | VUA All | | | | VUA Verb | | | | MOH-X (10 fold) | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Pre. | Rec. | F1 | Acc. | Pre. | Rec. | F1 | Acc. | Pre. | Rec. | F1 | Acc. |
| RNN_ELMo | 71.6 | 73.6 | 72.6 | 93.1 | 68.2 | 71.3 | 69.7 | 81.4 | 79.1 | 73.5 | 75.6 | 77.2 |
| RNN_BERT | 71.5 | 71.9 | 71.7 | 92.9 | 66.7 | 71.5 | 69.0 | 80.7 | 75.1 | 81.8 | 78.2 | 78.1 |
| RNN_HG | 71.8 | 76.3 | 74.0 | 93.6 | 69.3 | 72.3 | 70.8 | 82.1 | 79.7 | 79.8 | 79.8 | 79.7 |
| RNN_MHCA | 73.0 | 75.7 | 74.3 | 93.8 | 66.3 | **75.2** | 70.5 | 81.8 | 77.5 | 83.1 | 80.0 | 79.8 |
| MUL_GCN | 74.8 | 75.5 | 75.1 | 93.8 | 72.5 | 70.9 | 71.7 | 83.2 | 79.7 | 80.5 | 79.6 | 79.9 |
| RoBERTa_SEQ† | 80.4 | 74.9 | 77.5 | - | 79.2 | 69.8 | 74.2 | - | - | - | - | - |
| DeepMet† | *82.0* | 71.3 | 76.3 | - | *79.5* | 70.8 | 74.9 | - | - | - | - | - |
| MelBERT | 80.1 | *76.9* | *78.5* | - | 78.7 | 72.9 | *75.7* | - | - | - | - | - |
| MrBERT | **82.7** | 72.5 | 77.2 | *94.7* | **80.8** | 71.5 | **75.9** | **86.4** | *80.0* | **85.1** | *82.1* | *81.9* |
| MisNet | 80.4 | **78.4** | **79.4** | **94.9** | 78.3 | *73.6* | **75.9** | *86.0* | **84.2** | *84.0* | **83.4** | **83.6** |

Table 2: Results on VUA All, VUA Verb, and MOH-X. Best in **bold** and second best in *italic underlined*. The top block exhibits RNN based methods, while the middle block includes the transformer based. The † results are reproduced by Choi et al. (2021).[3]

detection and word sense disambiguation simultaneously. It also uses Graph Convolution Network with Bi-LSTM to encode dependency relations.

**RoBERTa_SEQ** (Leong et al., 2020): a sequence labeling baseline model provided by ACL 2020 metaphor detection shared task. RoBERTa_SEQ takes a sentence as input, and uses a softmax classifier to predict the metaphoricity for each token.

**DeepMet** (Su et al., 2020): the winning model in ACL 2020 metaphor detection shared task. It models metaphor identification as a reading comprehension task, with query features, fine-grained POS features, and context features etc. incorporated.

**MelBERT** (Choi et al., 2021): a model based on RoBERTa. It utilizes siamese network as well. However, MelBERT assumes that the target word used alone is literal.

**MrBERT** (Song et al., 2021): MrBERT regards metaphor detection as a relation classification task. It extracts dependency relations among verbs and subjects or objects, and embeds relations into BERT input.

### 4.3 Implementation Details

Gao et al. (2018) expanded VUA All dataset with POS tags. We retrieve basic usages from a digital Oxford dictionary[4] following Su et al. (2021). Since MOH-X does not have a training, validation, and test split, we perform 10-fold cross validation on it. Also, following previous studies (Choi et al., 2021; Song et al., 2021), we conduct zero-shot transfer on TroFi dataset to examine the generalization ability of MisNet. We use the RoBERTa (Liu et al., 2019) implementation for BERT, provided by HuggingFace[5]. It has stacked 12-layer transformer encoders, each with 12 attention heads. The hidden dimension in each layer is 768. Both hidden dimensions in MIP and SPV layer are 768.

**VUA All:** for VUA All dataset the learning rate is 3e-5. The epoch number is 15 and the training batch size is 64. Since VUA All suffers from data unbalance, we use different class weights in cross entropy loss function, 1 for literal samples and 5 for metaphors. **VUA Verb:** for VUA Verb, we remove POS tag from the input since it provides few information when only training on verbs. The training batch size is 64 with a 3e-5 learning rate. The class weights are 1 for literal samples and 4 for metaphors. We train for 15 epochs. **MOH-X:** MOH-X is a balanced dataset, such that we do not apply different class weights. The batch size is 16 with a 3e-5 learning rate, and we train for 15 epochs. All the experiments adopt AdamW (Peters et al., 2019) optimizer. For VUA All and VUA Verb, we take the best model on validation set to do testing. For MOH-X, we take the best score in each fold, and calculate the average over total 10 folds. All experiments are done in PyTorch 1.10 and cuda 11.2, on a single NVIDIA RTX 3090 GPU. Our code, saved model weights, and datasets are available for more details.

## 5 Results and Analysis

### 5.1 Overall Results

Following Mao et al. (2019), we mainly focus on F1 score. As shown in Table 2, MisNet obtains

[3] In ACL 2020 shared task, participants can manipulate training dataset or perform ensemble learning, making the original results incomparable.

[4] https://www.lexico.com/

[5] https://huggingface.co/roberta-base

| Model | Academic | | | | Conversation | | | | Fiction | | | | News | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Pre. | Rec. | F1 | Acc. | Pre. | Rec. | F1 | Acc. | Pre. | Rec. | F1 | Acc. | Pre. | Rec. | F1 | Acc. |
| RNN_ELMo | 78.2 | 80.2 | 79.2 | 92.8 | 64.9 | 63.1 | 64.0 | 94.6 | 61.4 | 69.1 | 65.1 | 93.1 | 72.7 | 71.2 | 71.9 | 91.6 |
| RNN_BERT | 76.7 | 76.0 | 76.4 | 91.9 | 64.7 | 64.2 | 64.4 | 94.6 | 66.5 | 68.6 | 67.5 | *93.9* | 71.2 | 72.5 | 71.8 | 91.4 |
| RNN_HG | 76.5 | **83.0** | 79.6 | 92.7 | 63.6 | **72.5** | 67.8 | *94.8* | 61.8 | 74.5 | 67.5 | 93.4 | 71.6 | *76.8* | 74.1 | 91.9 |
| RNN_MHCA | 79.6 | 80.0 | 79.8 | *93.0* | 64.0 | 71.1 | 67.4 | *94.8* | 64.8 | 70.9 | 67.7 | 93.8 | 74.8 | 75.3 | 75.0 | *92.4* |
| RoBERTa_SEQ† | *86.0* | 77.3 | 81.4 | - | 70.5 | 69.8 | 70.1 | - | 73.9 | 72.7 | 73.3 | - | 82.2 | 74.1 | *77.9* | - |
| DeepMet† | **88.4** | 74.7 | 81.0 | - | *71.6* | 71.1 | *71.4* | - | **76.1** | 70.1 | 73.0 | - | **84.1** | 67.6 | 75.0 | - |
| MelBERT | 85.3 | *82.5* | **83.9** | - | 70.1 | 71.7 | 70.9 | - | 74.0 | *76.8* | *75.4* | - | 81.0 | 73.7 | 77.2 | - |
| MisNet | 85.1 | *82.5* | *83.8* | **94.5** | **71.8** | *72.0* | **71.9** | **95.7** | *74.5* | **77.5** | **76.0** | **95.5** | *82.6* | **77.0** | **79.7** | **94.1** |

| Model | Verb | | | | Adjective | | | | Adverb | | | | Noun | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Pre. | Rec. | F1 | Acc. | Pre. | Rec. | F1 | Acc. | Pre. | Rec. | F1 | Acc. | Pre. | Rec. | F1 | Acc. |
| RNN_ELMo | 68.1 | 71.9 | 69.9 | - | 56.1 | 60.6 | 58.3 | - | 67.2 | 53.7 | 59.7 | 94.8 | 59.9 | 60.8 | 60.4 | - |
| RNN_BERT | 67.1 | 72.1 | 69.5 | 87.9 | 58.1 | 51.6 | 54.7 | 88.3 | 64.8 | 61.1 | 62.9 | 94.8 | 63.3 | 56.8 | 59.9 | 88.6 |
| RNN_HG | 66.4 | 75.5 | 70.7 | *88.0* | 59.2 | **65.6** | 62.2 | 89.1 | 61.0 | 66.8 | 63.8 | 94.5 | 60.3 | *66.8* | 63.4 | 88.4 |
| RNN_MHCA | 66.0 | *76.0* | 70.7 | 87.9 | 61.4 | 61.7 | 61.6 | *89.5* | 66.1 | 60.7 | 63.2 | *94.9* | 69.1 | 58.2 | 63.2 | *89.8* |
| RoBERTa_SEQ† | 74.4 | 75.1 | 74.8 | - | *72.0* | 57.1 | 63.7 | - | 77.6 | 63.9 | 70.1 | - | **76.5** | 59.0 | 66.6 | - |
| DeepMet† | **78.8** | 68.5 | 73.3 | - | **79.0** | 52.9 | 63.3 | - | *79.4* | 66.4 | 72.3 | - | **76.5** | 57.1 | 65.4 | - |
| MelBERT | 74.2 | 75.9 | *75.1* | - | 69.4 | 60.1 | *64.4* | - | **80.2** | *69.7* | **74.6** | - | *75.4* | 66.5 | **70.7** | - |
| MisNet | *77.5* | **77.7** | **77.6** | **91.4** | 68.8 | *65.2* | **67.0** | **91.2** | 76.4 | **70.5** | *73.3* | **96.3** | 74.4 | **67.2** | *70.6* | **91.6** |

Table 3: Breakdown results for genre and POS on VUA All. Best in **bold** and second best in *italic underlined*.

competitive results. In VUA All dataset, MisNet gains as most as 7.7 and 3.1 F1 score improvements compared with RNN based models and transformer based models respectively. Also, MisNet gains nearly 1.0 F1 score over the strongest baseline MelBERT, and achieves highest recall and accuracy scores. MisNet can fully utilize POS information, such that it has a strong ability to distinguish the impossible cases, like conjunctions and exclamations, from the potential ones.

In VUA Verb dataset, we remove POS tag in input because it provides little information when there is only one word class, i.e., we only make judgements via MIP and SPV layers. We get improvements by as most as 6.9 and 1.7 F1 scores compared with RNN based methods and transformer based methods respectively. It is worth mentioning that the strongest baseline MrBERT uses dependency parsing to extract subjects and objects for verbs, but MisNet still obtains promising results only via semantic matching methods, which shows the importance to properly utilize linguistic rules.

We attain improvements by 1.3 F1 scores against the strongest baseline MrBERT in MOH-X dataset, and achieve best precision and accuracy scores. Compared with RNN based methods, the performance is improved by as most as 7.8 F1 scores as well. We notice that MisNet performs better on MOH-X than VUA Verb: MOH-X is built upon WordNet via extracting metaphorical and literal usages of certain verbs, which means most metaphors in MOH-X are conventional metaphors. MisNet can get benefits from basic usages while the other baselines may fail to capture the basic meanings. However, verbs in VUA Verb dataset are much more complex, including auxiliary verbs, link verbs and etc. Predictions for VUA Verb are much harder.

## 5.2 VUA All Breakdown Results

Table 3 shows two breakdown analysis on VUA All dataset. In the genre track, MisNet outperforms the previous baselines in conversation, fiction, and news. We achieve a promising result on academic as well. All the methods perform better on academic and news, which have formal language usages so the patterns beneath are easy to perceive.

In the POS track, we find that MisNet achieves largest improvements on verb and adjective, with 2.5 and 2.6 F1 scores gained respectively. Verbs and adjectives are often used metaphorically, so there are more positive samples in VUA All dataset. Also, verb samples take the biggest portion in VUA All dataset, which makes the training on verbs more thorough. The performance on adverb is mediocre, because adverbs are very different internally. For instance, adverbs of time, place, and degree etc., can rarely be metaphors. Such complexity makes adverbs more difficult to judge.

### 5.3 Zero-shot Transfer on TroFi

We conduct zero-shot transfer on TroFi dataset, i.e., using TroFi only for testing. As Table 4 shows, MisNet outperforms all the baselines. Notably, the baselines with ♠ are trained on an expanded version of VUA All (Choi et al., 2021), so they have more training data. MrBERT explicitly utilizes dependency relations to benefit verb metaphor detection. However, we still attain best results in all metrics, which indicates that MisNet has strong generalization ability.

| **Model** | **TroFi (Zero-shot)** | | | |
|---|---|---|---|---|
| | Pre. | Rec. | F1 | Acc. |
| RoBERTa_SEQ♠ | 53.6 | 70.1 | 60.7 | - |
| DeepMet♠ | *53.7* | 72.9 | 61.7 | - |
| MelBERT♠ | 53.4 | 74.1 | 62.0 | - |
| MrBERT | **53.8** | *75.0* | *62.7* | *61.1* |
| MisNet | **53.8** | **76.2** | **63.1** | **61.2** |

Table 4: Zero-shot transfer results on TroFi dataset. We use MisNet trained on VUA All dataset.

### 5.4 Effectiveness Study

We conduct ablation experiments to test the effectiveness of different modules and features in MisNet, as Table 5 illustrates. In each ablation setting, the performance drops, which demonstrates the capability of each part. MIP module is more important than SPV as is observed. A conventional metaphor may be normal for its frequent context, so SPV becomes invalid. But MIP can notice the discrepancy between the contextual target meaning and its basic meaning. POS provides useful information for MisNet to filter out the impossible cases, without which the model performs worse. When basic usages are aborted, MisNet may fail to represent basic meanings, such that some metaphors cannot be detected. When feature embeddings are removed, MisNet works quite badly. Our designed feature embeddings can help model to treat different parts differently to better utilize features.

We also evaluate the impacts from different readout methods. We replace the readout method in Eq. 7 and Eq. 8 with candidates from Table 5. We find that $|\boldsymbol{u}-\boldsymbol{v}|$ and $(\boldsymbol{u};\boldsymbol{v})$ are two crucial components, without which the performance drops significantly. $|\boldsymbol{u}-\boldsymbol{v}|$ can directly reveal the difference between two representations, while $(\boldsymbol{u};\boldsymbol{v})$ can preserve all the original information. However, the default setting $(\boldsymbol{u};\boldsymbol{v};|\boldsymbol{u}-\boldsymbol{v}|;\boldsymbol{u}*\boldsymbol{v})$ is the best since all components work as an ensemble.

| **Ablation** | **Pre.** | **Rec.** | **F1** | **Acc.** |
|---|---|---|---|---|
| -MIP | **83.1** | 72.8 | 77.6 | 94.8 |
| -SPV | 81.2 | 76.0 | 78.5 | 94.8 |
| -POS | 79.1 | 77.4 | 78.2 | 94.6 |
| -Basic Usage | 81.2 | 75.4 | 78.2 | 94.8 |
| -Feature Embedding | 78.4 | 77.7 | 78.0 | 94.6 |
| MisNet♣ | 80.4 | **78.4** | **79.4** | **94.9** |
| **Readout Method** | **Pre.** | **Rec.** | **F1** | **Acc.** |
| $(\boldsymbol{u};\boldsymbol{v})$ | 81.5 | 76.1 | 78.7 | **94.9** |
| $(\lvert\boldsymbol{u}-\boldsymbol{v}\rvert)$ | **82.5** | 74.5 | 78.3 | **94.9** |
| $(\lvert\boldsymbol{u}*\boldsymbol{v}\rvert)$ | 73.9 | 80.4 | 77.1 | 94.0 |
| $(\lvert\boldsymbol{u}-\boldsymbol{v}\rvert;\boldsymbol{u}*\boldsymbol{v})$ | 75.5 | **81.2** | 78.3 | 94.4 |
| $(\boldsymbol{u};\boldsymbol{v};\boldsymbol{u}*\boldsymbol{v})$ | 82.4 | 74.2 | 78.1 | 94.8 |
| $(\boldsymbol{u};\boldsymbol{v};\lvert\boldsymbol{u}-\boldsymbol{v}\rvert)$ | 79.5 | 77.4 | 78.5 | 94.7 |
| $(\boldsymbol{u};\boldsymbol{v};\lvert\boldsymbol{u}-\boldsymbol{v}\rvert;\boldsymbol{u}*\boldsymbol{v})$♣ | 80.4 | 78.4 | **79.4** | **94.9** |

Table 5: Effectiveness study on VUA All dataset. ♣ are the default MisNet settings.

Table 6 shows the quality analysis results. The top block indicates that MisNet can better detect conventional metaphors by using basic usages, which confirms our assumptions at the beginning. The middle block includes indirect metaphors, of which the metaphoricity is predicted upon preceding words. Metaphors in the bottom block can be very confusing. If we do not use a wider context, we can't distinguish accurately. However, MisNet only takes sentence-level inputs, thus we cannot properly handle these situations. We leave it as a future work.

| **MisNet** | **-Basic U.** | **Label** | **Sentence** |
|---|---|---|---|
| ✓ | ✗ | M | The ban on emergency work was *tightened*. |
| ✓ | ✗ | M | A new minister would *operate* inside the DoE. |
| ✓ | ✗ | M | A financial *crash* of global proportions. |
| ✓ | ✗ | M | Raising the federal debt *ceiling*. |
| ✗ | ✗ | M | Er, just maybe the size of *this*. |
| ✗ | ✗ | M | I'm gonna play with *that* and see what. |
| ✗ | ✗ | M | She *bought* it. |
| ✗ | ✗ | M | Thought you might want a *lift*. |

Table 6: Quality analysis on VUA All dataset. Target words in *red italic*. **M** means **Metaphor**.

## 6 Conclusion

In this paper, we propose a novel metaphor detection model named MisNet, which uses MIP to compare the discrepancy between contextual target word meaning and its basic meaning, and utilizes

SPV to measure the incongruity between the target and its context. MisNet takes basic usages to encode basic target meanings, which can prevent the invalidation of MIP and SPV when dealing with conventional metaphors. Empirical results show that our method achieves competitive performance on several datasets.

## Acknowledgements

This work is supported by 2018 National Major Program of Philosophy and Social Science Fund of China (18ZDA238) and Beijing Social Science Fund (16YYB021).

## References

Dan Assaf, Yair Neuman, Yohai Cohen, Shlomo Argamon, Newton Howard, Mark Last, Ophir Frieder, and Moshe Koppel. 2013. Why "dark thoughts" aren't really dark: A novel algorithm for metaphor identification. pages 60–65.

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.

Julia Birke and Anoop Sarkar. 2007. Active learning for the identification of nonliteral language. In *Proceedings of the Workshop on Computational Approaches to Figurative Language*, pages 21–28, Rochester, New York. Association for Computational Linguistics.

Andreas Blank. 2013. *Why do new meanings occur? A cognitive typology of the motivations for lexical semantic change*. De Gruyter Mouton.

L. Bloomfield. 1994. *Language*. Motilal Banarsidass Publishers.

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.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680, Copenhagen, Denmark. Association for Computational Linguistics.

Peter Crisp, Raymond Gibbs, Alice Deignan, Graham Low, Gerard Steen, Lynne Cameron, Elena Semino, Joe Grady, Alan Cienki, Zoltan Kövecses, et al. 2007. Mip: A method for identifying metaphorically used words in discourse. *Metaphor and Symbol*, 22(1):1–39.

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.

Erik-Lân Do Dinh, Hannah Wieland, and Iryna Gurevych. 2018. Weeding out conventionalized metaphors: A corpus of novel metaphor annotations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1424, Brussels, Belgium. Association for Computational Linguistics.

John R Firth. 1957. A synopsis of linguistic theory, 1930-1955. *Studies in linguistic analysis*.

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.

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

Hessel Haagsma and Johannes Bjerva. 2016. Detecting novel metaphor using selectional preference information. In *Proceedings of the Fourth Workshop on Metaphor in NLP*, pages 10–17, San Diego, California. Association for Computational Linguistics.

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

Duong Le, My Thai, and Thien Nguyen. 2020. Multitask 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.

Chee Wee (Ben) 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, Online. Association for Computational Linguistics.

Zhenxi Lin, Qianli Ma, Jiangyue Yan, and Jieyu Chen. 2021. CATE: A contrastive pre-trained model for metaphor detection with semi-supervised learning.

In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3888–3898, Online and Punta Cana, Dominican Republic. 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.

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.

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.

Matthew E. Peters, Sebastian Ruder, and Noah A. Smith. 2019. To tune or not to tune? adapting pretrained representations to diverse tasks. In *Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019)*, pages 7–14, Florence, Italy. Association for Computational Linguistics.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Jinfeng Rao, Linqing Liu, Yi Tay, Wei Yang, Peng Shi, and Jimmy Lin. 2019. Bridging the gap between relevance matching and semantic matching for short text similarity modeling. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5370–5381, Hong Kong, China. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. 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.

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*, volume 14. John Benjamins Publishing.

Chang Su, Kechun Wu, and Yijiang Chen. 2021. Enhanced metaphor detection via incorporation of external knowledge based on linguistic theories. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1280–1287, Online. Association for Computational Linguistics.

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.

Xiaoyu Tong, Ekaterina Shutova, and Martha Lewis. 2021. Recent advances in neural metaphor processing: A linguistic, cognitive and social perspective. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4673–4686, 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, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Hai Wan, Jinxia Lin, Jianfeng Du, Dawei Shen, and Manrong Zhang. 2021. Enhancing metaphor detection by gloss-based interpretations. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1971–1981, Online. Association for Computational Linguistics.

Mingyu Wan, Kathleen Ahrens, Emmanuele Chersoni, Menghan Jiang, Qi Su, Rong Xiang, and Chu-Ren Huang. 2020. Using conceptual norms for metaphor detection. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 104–109, Online. Association for Computational Linguistics.

Yorick Wilks. 1975. A preferential, pattern-seeking, semantics for natural language inference. *Artificial intelligence*, 6(1):53–74.

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.

Runqi Yang, Jianhai Zhang, Xing Gao, Feng Ji, and Haiqing Chen. 2019. Simple and effective text matching with richer alignment features. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4699–4709, Florence, Italy. Association for Computational Linguistics.