Deep Learning-Based Knowledge Injection for Metaphor Detection: A Comprehensive Review

Anonymous ACL submission

Abstract

The history of metaphor research marks the evolution of knowledge infusion research. With the continued advancement of deep learning techniques in recent years, the natural language processing community has shown great interests in applying knowledge to successful results In metaphor detection task. Although there has been a gradual increase in the number of approaches involving knowledge injection in the field of metaphor detection, there is a lack of a complete review article on knowledge injection based approaches. Therefore, the goal of this paper is to provide a comprehensive review of research advances in the application of deep learning for knowledge injection In metaphor detection task. In this paper, we systematically summarize and generalize the mainstream knowledge and knowledge injection principles, as well as review the datasets, evaluation metrics, and benchmark models used In metaphor detection task. Finally, we explore the current issues facing knowledge injection methods and provide an outlook on future research.

1 Introduction

Metaphors are essentially cognitive mechanisms present in the human mind used to construct conceptual frameworks (Lakoff and Wehling, 2012). This phenomenon works by extracting familiar concepts in the target domain to understand vague and abstract concepts in the source domain (Lakoff and Johnson, 2008). As an important linguistic phenomenon, automatic detection of metaphors is crucial for many practical language processing tasks, including information extraction (Tsvetkov et al., 2013), sentiment analysis (Cambria et al., 2017), machine translation (Babieno et al., 2022), and seamless human-computer interaction (Rai and Chakraverty, 2021). In the philosophical account articulated in (Maloney, 1983), metaphor comprehension involves three distinct phases: comprehension of the literal interpretation, discovery of inconsistencies with the literal interpretation, and reasoning to recover the intended non-literal interpretation. It is important to emphasize that the focus of this study is to reveal the kinds and ways in which knowledge is integrated in a metaphor detection task, and does not deal with metaphor paraphrase generation. Therefore only the first two stages of metaphor comprehension will be explored.

Metaphor detection based on deep learning knowledge injection aims to fuse deep learning models and external knowledge to automatically identify metaphorical phenomena in text and improve model performance and generalization. Recently, researchers have been exploring the application of knowledge in metaphor detection. Mao et al. (2019) utilized generalized corpus information as the context with detected words using Metaphor Identification Program (MIP) (Group, 2007) and Selectional Preference Violation (SPV) (Wilks et al., 2013). Le et al. (2020) attempted to apply dependency tree knowledge to metaphor detection by constructing graph network adjacency matrices to utilize the dependency tree structural information. Su et al. (2020) used a cueing approach to transform metaphor detection into reading comprehension and introduced local text information. Choi et al. (2021) applied MIP and SPV to pre-trained models. Recently, Zhang and Liu (2023) achieved the state-of-the-art results for the current metaphor detection task by introducing semantic knowledge through adversarial learning and multi-task learning. However, current knowledge injection methods ignore the issue of the timeliness of metaphors, i.e., how to inject knowledge in different contexts and eras, what kind of knowledge to use, and the criteria for defining the meaning of metaphors. These issues are crucial for the metaphor detection task.

Although several research surveys on metaphor detection have existed in the past. Rai and Chakraverty (2021); Ptiček and Dobša (2023) provided an overview of metaphor theory and computational processing methods, Abulaish et al. (2020) surveyed six technical approaches to metaphorical language, and Tong et al. (2021) delved into metaphor processing methods and their applications. However, none of these surveys has taken the principle of knowledge infusion as a primary research focus. Against this background, our surveys aim to fill the gap in this research area. First, we systematically sorted out the mainstream knowledge methods and knowledge injection principles, and used an innovative categorization method to organically integrate these studies. Second, we conducted an exhaustive review and analysis of the current major metaphor datasets, including their different variants, assessment metrics, and benchmarks. Finally, we provided insights into the strengths and limitations of different knowledge injection methods, and offered suggestions and outlooks for future metaphor detection research.

2 Knowledge

In this section, we provide an introduction to the types of knowledge that are commonly used In metaphor detection task and how they are used.

2.1 Syntactic Knowledge

Part-of-Speech Tagging. Part-of-Speech (POS) is the tagging of each word in a sentence to indicate its grammatical role or lexical category in the context. Commonly used POS tag sets include Universal POS tag sets (Petrov et al., 2011), which defined a simplified set of lexical tokens with 17 tokens, such as NOUN (noun), VERB (verb), and Treebank tag sets (Santorini, 1990), which had more detailed tokens, including JJ (adjective), JJS (adjective with a supreme ending -est), etc. In metaphor detection task, researchers usually combine POS knowledge directly into the input sequence (Song et al., 2021; Feng and Ma, 2022), or construct multitask learning with POS as an auxiliary task (Le et al., 2020).

Dependency Tree. A Dependency Tree (DT) is a syntactic structural tree used to efficiently represent dependency relationships between words in a sentence. In a Dependency Tree, each word is given a node and is connected by edges to represent the directional relationship from the dependent (subordinate) word to its main dependent (head of the subordinate) word. In metaphor detection task, researchers often utilize dependency tree knowledge to improve the syntactic comprehension of their models. Le et al. (2020) employed Graph Convolutional Network (GCN), which used the dependency tree knowledge as an adjacency matrix to build a graphical structure of dependency relationships between words. Some studies (Song et al., 2021; Feng and Ma, 2022), on the other hand, focused on extracting subject-verb-object relationships in dependency trees to aid in metaphor detection. Song et al. (2021) processed the output of subject-predicate-object correspondences in text by combining, averaging, and maximizing to further capture the associations between structural semantics, while Feng and Ma (2022) used a BERT Decoder (Devlin et al., 2019) to allow the model to generate the start and end positions of subject-predicate-objects based on the context.

2.2 Semantic Knowledge

VerbNet. VerbNet (Schuler, 2005) is a verb categorization database containing nearly 4,000 English verb lemmas (lemma), and its category design refers to the study of Levin (Somers, 1994). In VerbNet, each verb is attributed to one or more categories that describe the semantic roles of the verb, syntactic constraints, and semantic relations between different categories, etc. VerbNet provides two main categorization approaches: based on syntactic structure and based on predicate meaning. In the metaphor detection task, researchers (Gong et al., 2020; Beigman Klebanov et al., 2016) used VerbNet's class information to convert each lexical unit into a binary feature vector.

FrameNet. The main goal of FrameNet (Baker et al., 1998; Lowe, 1997) is to provide sentences with semantic and syntactic annotations for a large part of the vocabulary in contemporary English. The corpus of this resource is built on The British National Corpus (Consortium et al., 2007). FrameNet employs a semantic description based on frames, each of which represents a semantic concept and describes the events, participants, attributes, relations, etc. associated with that concept. The project (Fillmore et al., 2002) is an extended version of FrameNet, which adds the US National Corpus resources. In the metaphor detection, Li et al. (2023c) used the FrameNet provided by (Fillmore et al., 2002) in the task for frame prediction of target and contextual lexical units, and the prediction results will aid in metaphorical analysis.

WordNet. WordNet (Miller, 1995; Fellbaum, 1998) is a hierarchically structured lexical database in which each word forms links with other related words to represent the semantic connections between them. In the metaphor detection task, Gong et al. (2020); Beigman Klebanov et al. (2016) classified words into fifteen categories based on the semantic links between words in WordNet and converted these categories into binary feature vectors. Such feature vectors can be used to assist the metaphor detection and improve the performance of the model. And Zhang and Liu (2023) considered the first of the WordNet example sentences as literal meanings and used it for multi-task learning.

Dictionary Knowledge. Dictionary example sentences or paraphrase knowledge are intended to provide the model with knowledge of the polysemous and metaphorical meanings of the words to be detected, and help the model better understand the semantic changes and metaphorical expressions of the words to be detected in different contexts. In metaphor detection, some researchers have utilized lexical examples to extract the context-based basic meanings of the words to be detected (Zhang and Liu, 2023), instead of the traditional approach of directly using the words to be detected as the basic meanings. Su et al. (2021) combined the lexical paraphrase information into the model input to achieve knowledge fusion.

Concreteness. Concreteness is the degree to which a word is characterized by the meaning it expresses in a language. In metaphor detection, researchers often relyed on the word specificity rating dataset (Brysbaert et al., 2014). This dataset was based on the SUBTLEX-US corpus (Brysbaert and New, 2009) and covers 37,058 token-level samples. This dataset was rated using a 5-point scale from abstract to concrete, and the data was collected with the help of Internet crowdsourcing. In previous studies (Klebanov et al., 2014; Gong et al., 2020; Beigman Klebanov et al., 2016), the lexical units to be detected were transformed into binary feature vectors depending on their specificity ratings.

Topic. Using the Latent Dirichlet Allocation (LDA) model (Blei et al., 2003), research scholars extracted a model containing 100 topics from the New York Times (NYT) corpus (Sandhaus, 2008) to characterize general topics discussed by the public. In the metaphor detection task, previous research work (Klebanov et al., 2014; Gong et al., 2020; Beigman Klebanov et al., 2016) matched and associated the words in each instance with these 100 topics, followed by the calculation of probability scores for each word under each topic.

2.3 Emotional Knowledge

VAD Model. VAD (Mehrabian, 1996) is an affective classification system for describing and measuring the three main dimensions of human affective experience: valence, arousal, and dominance. EmoBank corpus (Buechel and Hahn, 2017) is a VAD model-based and balanced multi-type 10k English corpus of sentences, each labeled with one to five ratings on the three VAD dimensions. In the metaphor detection task, Dankers et al. (2019) introduced the EmoBank corpus (Buechel and Hahn, 2017) as an auxiliary task. Sentence-level sentiment regression was constructed based on each dimension in EmoBank. In its training process, a batch of metaphor or sentiment task data sampling is randomly selected for training at each step.

Hyperbole Corpus. Exaggeration usually involves over- or under-exaggeration of an emotion, sentiment or attitude. Combining a dataset for hyperbole detection with a metaphor detection task can make the model more sensitive to capturing emotions and sentiments in text. In a previous research, Badathala et al. (2023) introduced two hyperbole corpora, named HYPO and HYPO-L, and subsequently labeled them with metaphors. The results showed that multitask learning based on hyperbole and metaphor gains in both two-way performance.

3 Method

This section will comprehensively introduce the current mainstream knowledge injection methods. Table 1 demonstrates a summary of knowledge injection-based metaphor detection systems.

3.1 Model Fine-tuning

Using semantic knowledge to fine-tune the model is a common approach. Li et al. (2023c) used two encoders, which first fine-tuned the Conceptual Encoder model on FrameNet Fillmore et al. (2002). For the output features $H = (h_{cls}, h_0, ..., h_n, h_{seq})$. We got the frame distribution of sentences and targets as follows:

$$\hat{y}_{cls} = sigmoid(W_0 h_{cls} + b_0) \quad (1)$$

$$\hat{y} = softmax(W_1 H + b_1), \quad (2)$$

where W_0 and W_1 are learnable parameters and b_0 and b_1 are biases. The fine-tuned Conceptual

SK	SYK	EK	Core Structure	Loss Function	Papers
✓			BERT+GRL / BERT+GBM	$\mathcal{L}=\frac{1}{\lvert\mathcal{N}^k\rvert}\sum_{i=1}^{\lvert\mathcal{N}^k\rvert}L_{ce}\left(\hat{y}_i^k,y_i^k\right)$	(Zhang and Liu, 2023) (Mao and Li, 2021) (Mao et al., 2022)
			Multi-task learning based on example sentence knowledge or POS labeling, where $\lvert\mathcal{N}^k\rvert$ represents the number of samples for the kth task, and $\hat{y}_i^k$ and y_i^k represent the predicted and true labeled values for the kth task, respectively.		
✓			BERT+MIP+SPV / BERT	$\mathcal{L}=\frac{1}{\lvert\mathcal{N}\rvert}\sum_{i=1}^{\lvert\mathcal{N}\rvert}L_{ce}\left(\hat{y}_i,y_i\right)$	(Li et al., 2023b) (Zhang and Liu) (Su et al., 2021) (Babieno et al., 2022)
			Using dictionary examples, and explanations of the words to be tested as auxiliary inputs.		
		✓	BERT	$\mathcal{L}_m=\frac{1}{\lvert\mathcal{N}\rvert}\sum_{i=1}^{\lvert\mathcal{N}\rvert}L_{ce}(\hat{y}_i,y_i)$; $\mathcal{L}_e=\sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i-y_i)^2}$	(Dankers et al., 2019) (Badathala et al., 2023)
			Introducing multi-task learning with affective knowledge, L_m for metaphor or exaggeration task loss, L_e for affective regression task loss.		
✓	✓		BERT	$\mathcal{L}^k=\sum_{i=1}^{\lvert\mathcal{N}\rvert}L_{ce}\left(\hat{s}_i^k,s_i^k\right)+\sum_{i=1}^{\lvert\mathcal{N}\rvert}L_{ce}\left(\hat{e}_i^k,e_i^k\right)$	(Feng and Ma, 2022)
			$k\in(sub,tar,obj)$ is the prediction loss for the main predicate, $\hat{s}_i^k$ and $\hat{e}_i^k$ denote the start and end positions of the kth grammatical category, respectively		
✓	✓		BERT	$\mathcal{L}=\frac{1}{\lvert\mathcal{N}\rvert}\sum_{i=1}^{\lvert\mathcal{N}\rvert}L_{ce}\left(\hat{y}_i,y_i\right)$	(Su et al., 2020) (Gong et al., 2020)
			Using semantic knowledge as portfolio input		
✓	✓		BiLSTM+GCN	$\mathcal{L}\left(x^t,y^t\right)=-\log P^t\left(y^t\mid x^t\right)+\lambda\left\lVert V^{wsd}-V^{md}\right\rVert_2^2$	(Le et al., 2020)
			Multi-task learning based on dependency trees, where $t\in wsd,md$. V^{wsd} and V^{md} denote the representation vectors of the same input sentence x^t, respectively.		
✓			BERT+MIP+SPV	$\mathcal{L}_k^c=-\frac{1}{\lvert\mathcal{N}\rvert}\sum_{i=1}^{\lvert\mathcal{N}\rvert}\sum_{j=1}^{\lvert\mathcal{C}\rvert}y_{ij}\log\left(\hat{y}_{ij}\right)$; $\mathcal{L}^s=-\frac{1}{\lvert\mathcal{N}\rvert}\sum_{i=1}^{\lvert\mathcal{N}\rvert}L_{ce}(\hat{y}_i,y_i)$	(Li et al., 2023c)
			$\mathcal{L}^c$ represents the loss of FrameNet fine-tuning encoder, where $k\in(target,cls)$ denotes the loss of global Frame and target word Frame; $\mathcal{L}^s$ denotes the loss of metaphorical classification.		
✓			BERT	$\mathcal{L}_1=\frac{1}{\lvert\mathcal{N}\rvert}\sum_{i=1}^{\lvert\mathcal{N}\rvert}L_{ce}(\hat{y}_i,y_i)$; $\mathcal{L}_2=\sum_{j=0}^{\lvert\mathcal{K}\rvert}I(f_i=g_i^j)\log(\alpha_i^j)$	(Wan et al., 2021)
			Multi-task learning for word sense disambiguation is introduced. Where f_i denotes the correct paraphrase of word $word_i$ in sentence s, $I(X)=1$ denotes when X is true, and $I(X)=0$ denotes when X is false.		
	✓		BERT+MIP+SPV / BERT	$\mathcal{L}=\frac{1}{\lvert\mathcal{N}\rvert}\sum_{i=1}^{\lvert\mathcal{N}\rvert}L_{ce}\left(\hat{y}_i,y_i\right)$	(Wang et al., 2023) (Song et al., 2021)
			Guiding the model to focus on structural information based on the dependency tree structure of the text.		

Table 1: Abstract of metaphor detection system based on knowledge injection. SK: semantic knowledge. SYK: syntactic knowledge. EK: emotional knowledge. core structure: subject model architecture.

Encoder will be injected with knowledge through MIP and SPV:

$$h_{MIP}=v_t\oplus v_{S,t}\oplus h_t\oplus h_{S,t} \tag{3}$$

$$h_{SPV}=v_{cls}\oplus v_{S,t}\oplus h_{cls}\oplus h_{S,t}, \tag{4}$$

where $v_{cls}, hcls$ denotes the hidden layer output corresponding to the two encoder CLS token inputs, v_t, h_t denote the isolated target word features, and $v_{S,t}, h_{S,t}$ denote the contextual target word features, respectively.

Le et al. (2020) used the structural knowledge from the dependency tree to build the adjacency matrix for the GCN network construction, which is denoted by the adjacency matrix A as:

$$A_{ij}=\begin{cases}1, & \text{if } i=j \text{ or } j\rightarrow i \text{ or } i\rightarrow j\\ 0, & \text{otherwise}\end{cases}. \tag{5}$$

3.2 Additional Inputs

This type of approach aims to enhance the model's understanding of the context by inputting knowledge into the model along with the text to be detected. The current dominant structure for knowledge-attached input are MIP and SPV.

MIP (Group, 2007) (Metaphor Identification Program) was originally introduced by the Pragglejaz Group. Its core logic consists in comparing the difference between a lexical unit in its original meaning and its meaning in context. For the text input $S = ([cls]w_0, ..., w_k, ...w_n[seq])$ and the test word w_k, the meaning of a lexical unit in context is defined as:

$$V_{S,k} = f_b(S)[k+1], \tag{6}$$

where f_b denotes the encoder and $V_{S,k}$ is the corresponding output of the kth hidden layer in the text to be detected. The vocabulary unit is intrinsically defined as:

$$V_k = f_b(w_k). \tag{7}$$

SPV (Selectional Preference Violation) was originally introduced to the metaphor detection (Wilks et al., 2013), and its core logic lied in comparing the semantic differences between lexical units and their surrounding words. The semantic information of surrounding words can be defined as:

$$V_S = f_b(S)[0], \tag{8}$$

where f_b denotes the encoder and V_S denotes the corresponding hidden layer output of cls in the text to be detected. And the semantic information of the SPV lexical units is similar to MIP as $V_{S,t}$.

In the metaphor detection task, the researcher introduced example sentences (Zhang and Liu; Li et al., 2023b), word paraphrases (Su et al., 2021; Babieno et al., 2022), or other relevant knowledge (Gong et al., 2020; Su et al., 2020) as additional input to the knowledge.

For the original sentence s and the word to be detected as w_k, Su et al. (2021) combines the first k paraphrases from the lexicon to the input and combines the output features through the mean-pool operation:

$$h^l = \begin{cases} f_b(w_k) & l = 0 \\ \frac{1}{N_l}\sum_{i=1}^{N_l} f_b(w_i^l) & l \neq 0 \end{cases}, \tag{9}$$

where f_b denotes the BERT encoder (Devlin et al., 2019), w_k is the target word, $l = 0$ is the original sentence, $l > 0$ is the paraphrase, and all the outputs will be used for the final classification after cat. While Zhang and Liu; Li et al. (2023b); Babieno et al. (2022) separate the sentence to be tested from the knowledge:

$$S_1 = ([cls]w_0, ...w_k, ...w_n[seq]), \tag{10}$$

$$S_2 = ([cls]w_0', ...w_k', ...w_n'[seq]), \tag{11}$$

where S_1, S_2 are the text to be detected and the knowledge text, respectively, and w_k' corresponds to the hidden layer output h_k' is the basic word paraphrase of the enhanced MIP.

3.3 Output Modulation

Pre-defined knowledge information can not only be used as input to the model, but also direct its attention to specific semantic content or syntactic structures when adjusting the model output. Wang et al. (2023); Song et al. (2021) assigned different weights to the model output based on the introduced knowledge.

Wang et al. (2023) measured the distance between the context word and the target word in terms of the number of jumps between neighboring words. For the output feature: $H = (h_1, h_2, \ldots, h_n)$, its final output is:

$$h_i' = \frac{1}{n}\sum h_i, i \in \mathcal{C}_n, \tag{12}$$

where h_i' is the final output feature and $\mathcal{C}_n, n \in (1, 2, 3, 4)$ denotes the nth neighboring word in the adjacent range of the parse tree centered on the target word.

Song et al. (2021) used three combination approaches (concatenation, average and maxout) to extract high dimensional features. For example, the contextual features obtained using the concatenation combination approach are defined as follows:

$$c = c_{subj} \oplus c_{obj} \oplus c_{cls} \oplus c_{bsc}, \tag{13}$$

where $c_k,\ k \in$ (cls, subj, obj, bsc), are the hidden layer outputs corresponding to the basic meanings of cls, subjects, objects and verbs, respectively. To capture the interaction of target lexical units with a specific context, Song et al. (2021) used linear, bilinear or a combination of both. Specifically, the linear and bilinear combination approaches are defined as follows, respectively:

$$r_{vc-linear} = \sigma(V_r \begin{pmatrix} v \\ c \end{pmatrix} + a_r), \tag{14}$$

$$r_{vc-bilinear} = \sigma(v^\intercal A_r c + b_r), \tag{15}$$

where V_r and A_r are the trainable weight parameters, respectively, and a_r and b_r are the bias parameters, σ is the sigmoid function.

Feng and Ma (2022) used decoder to allow the model to predict the position of the subject-predicate-object in the sentence, for the output text feature $H = (h_0, ..., h_n)$, there are:

$$h'_t = f_d(w'_{x<t}, H),\ t \in [1, 7], \tag{16}$$

where h'_t is the t th predicted output, $t \in [1, 6]$ corresponds to the indexes of the beginning and the end of the subject-predicate-object in the sentence, respectively, and h'_7 is the metaphorical classification result.

3.4 Multi-task Based Knowledge Fusion

Introducing other associated tasks can effectively promote knowledge fusion between tasks, thus helping to improve metaphor detection performance. In automatic metaphor detection, the researchers introduced several auxiliary tasks, including Word Sense Disambiguation (Le et al., 2020; Wan et al., 2021; Zhang and Liu, 2023), lexical annotation (Mao and Li, 2021; Mao et al., 2022), and tasks based on VAD sentiment labeling (Dankers et al., 2019).

For metaphor detection and lexical disambiguation input $s = (w_0, ..., w_n)$, $s' = (w'_0, ..., w'_n)$, Le et al. (2020) used different models to extract features with respectively:

$$H^{wsd} = f_b^{wsd}(s), H^{md} = f_b^{md}(s'), \tag{17}$$

where H^{wsd}, H^{md} are the word sense disambiguation and metaphor detection output features, respectively, and f_c^{wsd}, f_c^{md} are their corresponding models, and the output features will be used to obtain the fusion knowledge through similarity loss:

$$Loss_{similarity} = \lambda ||H^{wsd} - H^{md}||. \tag{18}$$

Wan et al. (2021) combined the two types of features by means of a combination:

$$p'_i = \sum_{k=0}^{m_i-1} \alpha_i^j p_i^j,\ \alpha_i^j = \frac{exp(h_i p_i^j)}{\sum_{k=0}^{m_i-1} exp(h_i p_i^j)}, \tag{19}$$

where h_i metaphorical text feature, p_i^j is the jth paraphrase of the ith metaphorical text, and the weighted combination of paraphrase features will be concatenated into the metaphorical features.

Zhang and Liu (2023) introduced the Gradient Reversal Layer (GRL) (Ganin and Lempitsky, 2015) module for adversarial learning. GRL aims to train on a large amount of labeled data in the source domain and a large amount of unlabeled data in the target domain through an inverse-gradient strategy, allowing the model to learn the target data distribution. Zhang and Liu (2023) constructed two lossy attempts to incorporate the knowledge of the WSD task into metaphor detection, for the i th input sample $s_i = (w_0, ..., w_n)$:

$$L^g = \frac{1}{|\mathcal{D}|} \sum L_{ce}((W^g f_b(s_i) + b^g), d_i), \tag{20}$$

$$L^l = \frac{1}{|\mathcal{D}|} \sum L_{ce}((W^l(\hat{y}_i f_b(s_i)) + b^l), d_i), \tag{21}$$

where f_b is the BERT encoder and $\hat{y}_i$ is the model output distribution. $\mathcal{D} = \mathcal{D}_{\mathcal{MD}} \cup \mathcal{D}_{\mathcal{WSD}}$, $d_i \in (0, 1)$ is the task labeling, where $0, 1$ are metaphor detection and WSD, respectively. Through an adversarial learning approach, the inverse gradient of L^g aims to steer the model for global knowledge fusion, while L^l further fine-tunes the output distribution of the model so that the metaphor-biased samples are closer to the non-basic samples in WSD, while the literal samples are closer to the basic samples, thus realizing fine-grained knowledge fusion.

Mao and Li (2021) proposed a gating mechanism based on multi-task learning, whose structure consists of three main parts: reset gate, update gate, and fusion gate. Considering the main task output feature H^j of a particular layer and multiple subtask features H^m (where $m \neq j$). The reset gate and its filtering features can be represented as:

$$R^m = \sigma(W_{\phi_R}^m H^m + b_{\phi_R}^m), \tag{22}$$

$$C^m = tanh(W_{\phi_C}^m (R^m \odot H^m) + b_{\phi_C}^m), \tag{23}$$

where $W_{\phi_R}^m, b_{\phi_R}^m, W_{\phi_C}^m, b_{\phi_C}^m$ is the trainable parameter, σ is the sigmoid activation function, and $\odot$ is the element-wise product. the update gate will learn joint information of the main task and a certain auxiliary task. For the main task H^j and a certain auxiliary task H^m, the update gate and its combined features are denoted as:

$$Z^m = \sigma(W_{\phi_Z}^m H^j + b_{\phi_Z}^m + V_{\phi_Z}^m C^m + d_{\phi_Z}^m), \tag{24}$$

$$F^m = Z^m \odot H^j + (1 - Z^m) \odot C^m, \tag{25}$$

where $W_{\phi_Z}^m, b_{\phi_Z}^m, V_{\phi_Z}^m, d_{\phi_Z}^m$ are the trainable parameters, and F^m are the combined features of the main task and the mth auxiliary task. The fusion gate will fuse all the combined features obtained by the update gate and use them for subsequent feature extraction or classification, i.e:

$$G^j = \sigma(W_{\phi_G}^j (\sum_{m \neq j} F^m) + b_{\phi_G}^j), \tag{26}$$

where $W_{\phi_G}^j, b_{\phi_G}^j$ are the trainable weights and bias.

4 Dataset and Metrics

The purpose of this section is to provide an overview of the current mainstream metaphor detection datasets, details about which have been listed in Table 2. Also, we will introduce the evaluation metrics commonly used in this field (see Appendix A for details). In addition, we will also summarize the performance of metaphor detection tasks performed on the four datasets VUA ALL, VUA verb, MOH-X and TroFi in recent years (see Table 3 in Appendix B for more details), to present a comprehensive picture of the state of the art of research in this area.

VUAMC. The VU Amsterdam Metaphor Corpus (Steen et al., 2010) annotated each lexical unit (187,570 in total) in a subset of the British National Corpus (Consortium et al., 2007) metaphorically. The corpus tags sentences used the MIPVU metaphor detection program. VUAMC is the largest publicly available annotated corpus for token-level metaphor detection, and the only one that investigates the metaphorical nature of dummy words. Based on VUAMC, many variants of VUA have emerged.

VUA ALL POS. VUA ALL POS dataset has been applied to the shared task of metaphor detection (Leong et al., 2018, 2020), which consists of two parts, VUA ALL POS and VUA Verb. In particular, VUA ALL POS annotates all real-sense words (including adjectives, verbs, nouns, and adjectives) in a sentence; while VUA Verb covers only verbs. However, in the studies of (Song et al., 2021; Feng and Ma, 2022; Wan et al., 2021; Su et al., 2020), the VUA ALL POS dataset also includes dummy words. To distinguish it from the shared task (Leong et al., 2018, 2020), we named the VUA ALL POS dataset that includes both real and dummy words as VUA ALL.

VUA Verb. Since VUA Verb contains only verbs and no other variants, we used the dataset reported in the metaphor detection shared task (Leong et al., 2018, 2020). In VUA Verb, 15,516 samples were used for training and 5,873 for testing.

VUA SEQ. VUA SEQ is another dataset constructed based on VUAMC. Compared to VUA ALL, VUA SEQ has the same number of samples as reported (Gao et al., 2018; Neidlein et al., 2020). However, VUA SEQ covers all tokens in a sentence, even punctuation, in the classification task, thus leading to a richer number of target tokens used than VUA ALL.

VUA18. According to the research (Choi et al., 2021), VUA-18 is very similar to VUA-SEQ and VUA ALL as they use the same sentences in each subset, 6,323, 1,550, and 2,694 sentences for the training, development, and test sets, respectively. VUA-18 does not consider abbreviations and punctuation as separate tokens, and has the same labeling rules as VUA ALL. We therefore categorized VUA-18 with VUA ALL.

VUA20. In the literature (Choi et al., 2021; Li et al., 2023c; Wang et al., 2023), VUA20 labeled 1.2k sentences with real and imaginary words. However, this did not match the description in the 20-year shared task (Leong et al., 2020). The text stated that it uses the same VUA as the 18-year shared task (Leong et al., 2018) (see Section 3.1, lines 8-10) and that both report the same token count.

TroFi. TroFi is a verb-target focused dataset containing the literal and metaphorical usage of 50 English verbs from the 1987-1989 Wall Street Journal corpus (Charniak et al., 2000). The dataset contains a total of 3717 samples, including 2741 training samples and 968 test samples.

MOH. The MOH dataset (Mohammad et al., 2016) consists of 1639 sentences extracted from WordNet, which contains 1230 sentences for literal usage and 409 sentences for metaphorical usage. And it was labeled with metaphors using crowdsourcing. MOH-X (Shutova et al., 2016), on the other hand, is a subset of the MOH dataset that focuses on collecting samples containing verbs. In MOH-X, each verb covers multiple semantic meanings, at least one of which is metaphorical usage.

Dataset	#Tok.	#Sent.	%Met.
VUAall/SEQ	205,425	10,567	11.6%
VUAall/SEQ/tr	116,622	6,323	11.2%
VUAall/SEQ/val	38,628	1,550	11.6%
VUAall/SEQ/te	50,175	2,694	12.4%
VUAallpos	94,807	16,202	15.8%
VUAallpos_tr	72,611	12,122	15.2%
VUAallpos_te	22,196	4,080	17.9%
VUAverb_tr	15,516	7,479	27.9%
VUAverb_val	1,724	1,541	26.9%
VUAverb_te	5,873	2,694	29.9%
MOH-X	647	647	48.7%
TroFi	3,737	3,737	43.5%

Table 2: tr: training set. val: validation set. te: test set. tokens: number of samples. sent.: total number of sentences, %Met.: percentage of metaphorical samples

5 Conclusion

In this section, we summarize the problems faced in the metaphor detection task as follows and explore possible future research directions.

Refining the Criteria for Defining Metaphors. Current research ignores the problem of lexical lag. Lexical lag refers to the lack of a clear basis for researchers to define literal meanings when introducing knowledge. For example, some studies consider the first paraphrase in WordNet as the basic meaning Zhang and Liu (2023), or use the first k example sentences in the dictionary as the criterion for classifying non-metaphorical expressions (Su et al., 2021; Zhang and Liu). However, these approaches may lead the model to incorrectly interpret metaphorical words as literal meanings. Therefore, there is a need for continuous refinement of the metaphor definition criteria to improve the accuracy of knowledge incorporation.

Enhancing the Knowledge Infusion Methodology. Most of the past studies injected knowledge directly into the inputs of the model (Li et al., 2023b; Babieno et al., 2022) or adjusted the outputs (Wang et al., 2023; Feng and Ma, 2022). However, using this combination alone may not be able to fully utilize the rich contextual information in the knowledge. To improve the efficiency of knowledge injection, Le et al. (2020) used dependency tree structure information to construct the adjacency matrix of Graph Convolutional Network (GCN), Li et al. (2023c) fine-tuned the model with FrameNet to capture the implicit knowledge, and Mao and Li (2021); Mao et al. (2022) designed a gating mechanism for extracting the associations between the main task and several subtasks separately information. Although there have been some attempts to improve the knowledge injection approach, this area is still an active research direction.

Exploring Fine-Grained Emotions. Many studies have shown that there is a close connection between textual emotions and metaphors (Mohammad et al., 2016; Li et al., 2023a). Among the previous researches, Dankers et al. (2019) designed an emotion-based VAD labeling sentiment regression task. While Badathala et al. (2023) skillfully introduced a hyperbole corpus to realize the bidirectional efficiency of hyperbole and metaphor detection performance. It is worth noting that the above studies are limited to rough sample combination in terms of knowledge fusion methods, and have not yet explored in-depth more detailed ways of emotion knowledge injection, such as at the lexical level or at the level of sentence structure.

Exploring Zero-shot Metaphor Detection. In view of the resource overhead problem associated with supervised metaphor detection, the zero-shot metaphor detection has been attempted by some scholars in past researches. Among them, Mao et al. (2018) introduced WordNet's superordinate and synonyms, by calculating the cosine distance between the context and the target word, if it is greater than a set threshold, it is determined to be a metaphor. Mao et al. (2022) adopted a similar approach, where the proximity word is selected from the candidate set, i.e., the word with the highest probability of occurrence in the BERT context. The above methods provided meaningful explorations in the zero-shot metaphor detection for knowledge injection and some insights for future research.

Introducing Multilingual Knowledge An important but little explored research direction is the construction of multilingual metaphor detection models. Tsvetkov et al. (2013, 2014) have constructed cross-language metaphor detection models by training them on English samples and applying them to the target language. Sanchez-Bayona and Agerri (2022) constructed CoMeta, the first corpus annotated with Spanish metaphors, and designed two zero-shot experiments using CoMeta and VUA (Steen et al., 2010) as the training and test sets, respectively, thus demonstrating cross-linguistic consistency between languages. These studies provide the feasibility and value of exploring cross-lingual knowledge injection.

6 Limitations

This paper provides a comprehensive description of metaphor detection systems in deep learning, focusing on discussing and summarizing in detail the different types and methods of model knowledge injection. However, there exists a small amount of research work in the area of metaphor detection that does not use knowledge or employs unsupervised methods, and these studies are not covered or discussed in the paper. In future research, we plan to provide a comprehensive summary of most of the work in the area of metaphor detection, including both supervised and unsupervised approaches, to provide researchers with a more comprehensive understanding.

7 Ethics Statement

In this paper, we provide a detailed description of the supervised metaphor detection system and the different ways of knowledge injection. The datasets and research papers we have used have been obtained from publicly available sources and we have adhered to strict guidelines of academic and research ethics. In addition, we place special emphasis on transparency and openness of information, encourage other researchers to conduct responsible research, and uphold best practices in knowledge sharing. In the text, we explicitly cite the public data sources cited to express our full respect for the original authors and data providers of research related to the field of metaphor detection.

References

Muhammad Abulaish, Ashraf Kamal, and Mohammed J. Zaki. 2020. A survey of figurative language and its computational detection in online social networks. *ACM Transactions on the Web*, 14(1):1–52.

Mateusz Babieno, Masashi Takeshita, Dusan Radisavljevic, Rafal Rzepka, and Kenji Araki. 2022. Miss roberta wilde: Metaphor identification using masked language model with wiktionary lexical definitions. *Applied Sciences*, 12(4):2081.

Naveen Badathala, AbisekRajakumar Kalarani, Tejpalsingh Siledar, and Pushpak Bhattacharyya. 2023. A match made in heaven: A multi-task framework for hyperbole and metaphor detection.

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The berkeley framenet project. In *Proceedings of the 36th annual meeting on Association for Computational Linguistics -*.

Beata Beigman Klebanov, Chee Wee Leong, E. Dario Gutierrez, Ekaterina Shutova, and Michael Flor. 2016. Semantic classifications for detection of verb metaphors. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Marc Brysbaert and Boris New. 2009. Moving beyond kučera and francis: A critical evaluation of current word frequency norms and the introduction of a new and improved word frequency measure for american english. *Behavior Research Methods*, page 977–990.

Marc Brysbaert, Amy Beth Warriner, and Victor Kuperman. 2014. Concreteness ratings for 40 thousand generally known english word lemmas. *Behavior Research Methods*, 46(3):904–911.

Sven Buechel and Udo Hahn. 2017. Emobank: Studying the impact of annotation perspective and representation format on dimensional emotion analysis. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*.

Erik Cambria, Soujanya Poria, Alexander Gelbukh, and Mike Thelwall. 2017. Sentiment analysis is a big suitcase. *IEEE Intelligent Systems*, 32(6):74–80.

Eugene Charniak, Don Blaheta, Niyu Ge, Keith Hall, John Hale, and Mark Johnson. 2000. Bllip 1987-89 wsj corpus release 1. *Linguistic Data Consortium, Philadelphia*, 36.

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*.

BNC Consortium et al. 2007. British national corpus. *Oxford Text Archive Core Collection*.

Verna Dankers, Marek Rei, Martha Lewis, and Ekaterina Shutova. 2019. Modelling the interplay of metaphor and emotion through multitask learning. 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)*.

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*.

Christiane Fellbaum. 1998. *WordNet: An electronic lexical database*. MIT press.

Huawen Feng and Qianli Ma. 2022. It's better to teach fishing than giving a fish: An auto-augmented structure-aware generative model for metaphor detection. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 656–667.

CharlesJ. Fillmore, CollinF. Baker, and Hiroaki Sato. 2002. The framenet database and software tools. *Language Resources and Evaluation,Language Resources and Evaluation*.

Yaroslav Ganin and Victor Lempitsky. 2015. Unsupervised domain adaptation by backpropagation. *International Conference on Machine Learning,International Conference on Machine Learning*.

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*.

Hongyu Gong, Kshitij Gupta, Akriti Jain, and Suma Bhat. 2020. Illinimet: Illinois system for metaphor detection with contextual and linguistic information. In *Proceedings of the Second Workshop on Figurative Language Processing*.

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

Beata Beigman Klebanov, Ben Leong, Michael Heilman, and Michael Flor. 2014. Different texts, same metaphors: Unigrams and beyond. In *Proceedings of the Second Workshop on Metaphor in NLP*, pages 11–17.

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

George Lakoff and Elisabeth Wehling. 2012. *The little blue book: The essential guide to thinking and talking democratic*. Simon and Schuster.

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

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*.

Chee Wee (Ben) 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*.

Yucheng Li, Frank Guerin, and Chenghua Lin. 2023a. The secret of metaphor on expressing stronger emotion.

Yucheng Li, Shun Wang, Chenghua Lin, and Guerin Frank. 2023b. Metaphor detection via explicit basic meanings modelling.

Yucheng Li, Shun Wang, Chenghua Lin, Frank Guerin, and Loïc Barrault. 2023c. Framebert: Conceptual metaphor detection with frame embedding learning.

Zhenxi Lin, Qianli Ma, Jiangyue Yan, and Jieyu Chen. 2021. Cate: A contrastive pre-trained model for metaphor detection with semi-supervised learning. *Empirical Methods in Natural Language Processing,Empirical Methods in Natural Language Processing*.

JohnB. Lowe. 1997. A frame-semantic approach to semantic annotation.

J. Christopher Maloney. 1983. A new model for metaphor. *Dialectica*, 37(4):285–301.

Rui Mao and Xiao Li. 2021. Bridging towers of multi-task learning with a gating mechanism for aspect-based sentiment analysis and sequential metaphor identification. *Proceedings of the ... AAAI Conference on Artificial Intelligence,Proceedings of the ... AAAI Conference on Artificial Intelligence*.

Rui Mao, Xiao Li, Mengshi Ge, and Erik Cambria. 2022. Metapro: A computational metaphor processing model for text pre-processing. *Information Fusion*, 86–87:30–43.

Rui Mao, Chenghua Lin, and Frank Guerin. 2018. Word embedding and wordnet based metaphor identification and interpretation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

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*.

Albert Mehrabian. 1996. Pleasure-arousal-dominance: A general framework for describing and measuring individual differences in temperament. *Current Psychology*, page 261–292.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

SaifM. Mohammad, Ekaterina Shutova, and PeterD. Turney. 2016. Metaphor as a medium for emotion: An empirical study. *Joint Conference on Lexical and Computational Semantics,Joint Conference on Lexical and Computational Semantics*.

Arthur Neidlein, Philip Wiesenbach, and Katja Markert. 2020. An analysis of language models for metaphor recognition. In *Proceedings of the 28th International Conference on Computational Linguistics*.

Slav Petrov, Dipanjan Das, and Ryan McDonald. 2011. A universal part-of-speech tagset. *Cornell University - arXiv,Cornell University - arXiv*.

Martina Ptiček and Jasminka Dobša. 2023. Methods of annotating and identifying metaphors in the field of natural language processing. *Future Internet*, 15(6):201.

Sunny Rai and Shampa Chakraverty. 2021. A survey on computational metaphor processing. *ACM Computing Surveys*, page 1–37.

Omid Rohanian, Marek Rei, Shiva Taslimipoor, and Le An Ha. 2020. Verbal multiword expressions for identification of metaphor. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

Elisa Sanchez-Bayona and Rodrigo Agerri. 2022. Leveraging a new spanish corpus for multilingual and crosslingual metaphor detection.

Evan Sandhaus. 2008. The new york times annotated corpus.

Beatrice Santorini. 1990. Part-of-speech tagging guidelines for the penn treebank project (3rd revision).

Karin Kipper Schuler. 2005. *VerbNet: A broad-coverage, comprehensive verb lexicon*. University of Pennsylvania.

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

HaroldL. Somers. 1994. Book reviews: English verb classes and alternations: A preliminary investigation.

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)*.

Gerard J. Steen, Aletta G. Dorst, J. Berenike Herrmann, Anna Kaal, Tina Krennmayr, and Trijntje Pasma. 2010. A method for linguistic metaphor identification.

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*.

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*.

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*.

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)*.

Yulia Tsvetkov, Elena Mukomel, and Anatole Gershman. 2013. Cross-lingual metaphor detection using common semantic features.

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*.

Shun Wang, Yucheng Li, Chenghua Lin, Loïc Barrault, and Frank Guerin. 2023. Metaphor detection with effective context denoising.

Yorick Wilks, Adam Dalton, JamesF. Allen, and Lucian Galescu. 2013. Automatic metaphor detection using large-scale lexical resources and conventional metaphor extraction.

Shenglong Zhang and Ying Liu. Metaphor detection via linguistics enhanced siamese network.

Shenglong Zhang and Ying Liu. 2023. Adversarial multi-task learning for end-to-end metaphor detection. *arXiv preprint arXiv:2305.16638*.

A Metrics

Current mainstream neural models treated metaphor detection as a dichotomous sequence labeling task. Among their commonly used evaluation metrics, precision measures the degree of correct prediction, while recall measures the completeness of the categorization or information retrieval system. The reconciled mean of precision and recall is known as the F-score, which is high when both precision and recall are high. Precision, Recall, F-score and Accuracy are defined respectively:

$$Pre = \frac{TP}{TP + FP} \tag{27}$$

$$Rec = \frac{TP}{TP + FN} \tag{28}$$

$$F1 = \frac{2 \times Pre \times Rec}{Pre} \tag{29}$$

$$Acc = \frac{TP + TN}{TP + TN + FP + FN}, \tag{30}$$

where True Positives (TP) denote the number of texts recognized as metaphorical and actually metaphorical, False Positives (FP) denote the number of texts recognized as metaphorical but actually non-metaphorical, True Negatives (TN) denote the number of texts recognized as non-metaphorical and actually non-metaphorical, and False Negatives (FN) denote the number of texts recognized as non-metaphorical but actually metaphorical.

B Model Performance

	VUA ALL				VUA Verb				MOH-X (10 fold)				TroFi (10 fold)			
	Pre.	Rec.	F1	Acc.	Pre.	Rec.	F1	Acc.	Pre.	Rec.	F1	Acc.	Pre.	Rec.	F1	Acc.
(Gao et al., 2018)					53.4	65.6	58.9	69.1	75.3	84.3	79.1	78.5	68.7	74.6	72	73.7
(Gao et al., 2018)	71.6	73.6	72.6	93.1	68.2	71.3	69.7	81.4	79.1	73.5	75.6	77.2	70.1	71.6	71.1	74.6
(Gao et al., 2018)	71.5	71.9	71.7	92.9	66.7	71.5	69	80.7	75.1	81.8	78.2	78.1	70.3	67.1	68.7	73.4
(Mao et al., 2019)	71.8	76.3	74	93.6	69.3	72.3	70.8	82.1	79.7	79.8	79.8	79.7	67.4	77.8	72.2	74.9
(Mao et al., 2019)	73	75.7	74.3	93.8	66.3	75.2	70.5	81.8	77.5	83.1	80	79.8	68.6	76.8	72.4	75.2
(Gong et al., 2020)	74.6	71.5	73		76.7	77.2	77						72.6	67.5	69	
(Le et al., 2020)	74.8	75.5	75.1		72.5	70.9	71.7	83.2	79.7	80.5	79.6	79.9	73.1	73.6	73.2	76.4
(Rohanian et al., 2020)									80	80.4	80.2	80.5	73.8	71.8	72.8	73.5
(Leong et al., 2020)	80.4	74.9	77.5		79.2	69.8	74.2									
(Su et al., 2020)	82	71.3	76.3		79.5	70.8	74.9		79.9†	76.5†	77.9†		53.7†	72.9†	61.7†	
(Song et al., 2021)	82.7	72.5	77.2	94.7	80.8	71.5	75.9	86.4	80	85.1	82.1	81.9	70.4	74.3	72.2	75.1
(Wan et al., 2021)	82.5	72.5	77.2	94.7	78.9	70.9	74.7	85.4								
(Choi et al., 2021)	80.1	76.9	78.5		78.7	72.9	75.7		79.3†	79.7†	79.2†		53.4†	74.1†	62†	
(Li et al., 2023c)	82.7	75.3	78.8						83.2†	84.4†	83.8†		70.7†	78.2†	74.2†	
(Babieno et al., 2022)	79.3	78.5	78.9		60.9	77.7	68.3		81	80	80.2		53.2	72.8	61.4	
(Lin et al., 2021)	79.3	78.8	79	94.8	78.1	73.2	75.6	85.8	85.7	84.6	84.7	85.2	74.4	74.8	74.5	77.7
(Wang et al., 2023)	80	78.2	79.1						77*	83.5*	80.1*		54.2*	76.2*	63.3*	
(Zhang and Liu)	80.4	78.4	79.4	94.9	78.3	73.6	75.9	86	84	84	83.4	83.6	67.5	77.6	71.9	73.6
(Feng and Ma, 2022)	81.6	77.4	79.4	95.2	81.6	71.1	76	86.4	89.5	85.2	87	87.5	72.5	77.5	74.8	77.7
(Su et al., 2021)					76	76	76	85.7	82.9	84	83.4	84.2	73.3	69.6	71.4	75.7
(Zhang and Liu, 2023)	78.4	79.5	79	94.7	78.5	78.1	78.3	87	87.4	88.8	87.9	88	70.5	79.8	74.7	76.5

Table 3: This table shows the performance of the metaphor detection system on four datasets, VUA ALL, VUA verb, MOH-X and TroFi, in recent years. Among them, most of the results on the MOH-X and TroFi datasets are based on ten-fold cross-validation, and also include some results derived from direct computation († labeling), as well as some of the models are trained on the VUA20 dataset (* labeling).