Bridging Word-Pair and Token-Level Metaphor Detection with Explainable Domain Mining

Yuan Tian[1,2], Ruike Zhang[1,2], Nan Xu[1,3], Wenji Mao[1,2]*
[1]State Key Laboratory of Multimodal Artificial Intelligence Systems,
Institute of Automation, Chinese Academy of Sciences
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences
[3]Beijing Wenge Technology Co., Ltd
{tianyuan2021,zhangruike2020,xunan2015,wenji.mao}@ia.ac.cn

Abstract

Metaphor detection aims to identify whether a linguistic expression in text is metaphorical or literal. Most existing research tackles this problem either using word-pair or token-level information as input, and thus treats word-pair and token-level metaphor detection as distinct subtasks. Benefited from the simplified structure of word pairs, recent methods for word-pair metaphor detection can provide intermediate explainable clues for the detection results, which remains a challenging issue for token-level metaphor detection. To mitigate this issue in token-level metaphor detection and take advantage of word pairs, in this paper, we make the first attempt to bridge word-pair and token-level metaphor detection via modeling word pairs within a sentence as explainable intermediate information. As the central role of verb in metaphorical expressions, we focus on token-level verb metaphor detection and propose a novel explainable **W**ord **P**air based **D**omain **M**ining (WPDM) method. Our work is inspired by conceptual metaphor theory (CMT). We first devise an approach for conceptual domain mining utilizing semantic role mapping and resources at cognitive, commonsense and lexical levels. We then leverage the inconsistency between source and target domains for core word pair modeling to facilitate the explainability. Experiments on four datasets verify the effectiveness of our method and demonstrate its capability to provide the core word pair and corresponding conceptual domains as explainable clues for metaphor detection.

1 Introduction

Metaphor is not just a figurative expression but a pervasive phenomenon in human thought, perception, and reasoning (Lakoff and Johnson, 1980). As defined in Merriam-Webster Dictionary, *metaphor* is "a figure of speech in which a word/phrase literally denoting one kind of object or idea is used in place of another to suggest a likeness or analogy between them". Metaphor detection, as a fundamental research task in natural language processing (NLP), focuses on distinguishing metaphorical expressions from literal expressions in text. It can benefit a variety of other NLP tasks, which require the understanding of implicit semantics, such as machine translation (Mao et al., 2018), sentiment analysis (Mao and Li, 2021), and conversational dialogue (Sun et al., 2023).

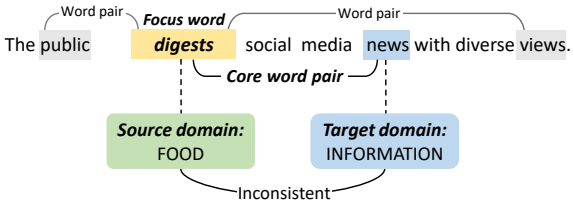

Figure 1: Illustration of token-level metaphor understanding with the core word pair and corresponding source and target domains. The focus word *digests* in the sentence is a token-level metaphor. In this sentence, there are several candidate context words that can form word pairs with the focus word. The core word pair *digest news* and its implicit source and target domains can help metaphor detection and explanation.

Previous studies on metaphor detection mainly focus on two distinct subtasks, word-pair metaphor detection (Ge et al., 2022; Tian et al., 2023) and token-level metaphor detection (Choi et al., 2021; Li et al., 2023). The former considers word pair as the fundamental unit conveying metaphorical meaning and classifies word pairs into metaphorical or literal categories, while the latter identifies words within sentences that imply metaphorical meaning. For example, in Figure 1, *digests news* can be classified as a metaphorical word pair and the word *digests* within the sentence is a token-level metaphor conveying metaphorical meaning.

Early research on word-pair metaphor detection employs machine learning or deep learning methods based on linguistic features related to metaphor

*Corresponding author

Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 13311–13325

(Tsvetkov et al., 2014; Shutova et al., 2016; Rei et al., 2017). Recent research is inspired by conceptual metaphor theory (CMT) (Lakoff and Johnson, 1980), which reveals the underlying process of metaphor understanding in human cognition and argues that a metaphor implies the inconsistency between source and target domains. These works (Ge et al., 2022; Tian et al., 2023) propose methods to model the intermediate explainable domain information for better word-pair metaphor detection.

To detect token-level metaphors, early studies employ statistical or RNN-based methods (Li et al., 2013; Wu et al., 2018; Mao et al., 2019). Most recent studies (Choi et al., 2021; Zhang and Liu, 2022; Li et al., 2023) have developed methods based on the elements and relations described in metaphor identification theories, including selection preference violation (SPV) (Wilks, 1975, 1978) and metaphor identification procedure (MIP) (Pragglejaz Group, 2007).

Benefited from the simplified structure of word pairs, Ge et al. (2022) can capture intermediate explainable clues including source and target domains to enhance word-pair metaphor detection. However, modeling such explainable information for better detection remains a challenging issue for token-level metaphor detection. As token-level metaphor detection involves complex semantic structure and sentence-level context, it brings greater research challenges for both detection and explanation compared to word-pair metaphor detection. Moreover, token-level and word-pair metaphor detection are treated as distinct subtasks in existing research. The intermediate information and explainable clues that can be derived from word pairs have never been explored in previous token-level research, nor are they properly utilized in the token-level detection and explanation processes. For example, Figure 1 illustrates a token-level metaphor, indicating that the core word pair and corresponding source and target domains can serve as valuable clues for understanding and detecting token-level metaphors.

In this paper, we take the first step to bridge word-pair and token-level metaphor detection and propose an explainable **W**ord **P**air based **D**omain **M**ining (WPDM) method for token-level verb metaphor detection. Since in a token-level metaphorical sentence, there are several candidate context words that can form word pairs with the focus word, it is non-trivial to identify the core word pair that conveys the primary metaphorical meaning. Thus, inspired by CMT, we mine the domain concepts associated with each word pair utilizing semantic role correspondence and domain granularity assessment for core word pair selection. Specifically, we first devise an approach to mine source and target domain information via mapping semantic roles based on VerbNet (Schuler, 2005) and leveraging domain concepts from cognitive, commonsense and lexical knowledge resources. We then exploit the inconsistency between conceptual source and target domains to determine core word pair attentions for facilitating explainable metaphor detection. The main contributions of our work are as follows:

- We make the first attempt to connect word-pair and token-level metaphor detection for more fine-grained identification and understanding of metaphors at the sentence level.
- To facilitate explainable token-level metaphor detection, we propose a novel word pair based domain mining method inspired by CMT, which consists of semantic role mapping and conceptual domain mining for core word pair modeling based on cognitive, commonsense and lexical resources.
- Extensive experiments on four datasets verify the effectiveness of our method and also demonstrate its capability to identify the core word pair and corresponding conceptual domains as explainable results for token-level metaphor detection.

2 Related Work

Metaphor detection aims to distinguish metaphorical and literal expressions in text. Existing studies focus on two subtasks: word-pair metaphor detection and token-level metaphor detection. The former determines whether a word pair is metaphorical or literal, and the latter concentrates on identifying metaphorical words within sentences.

Word-Pair Metaphor Detection Traditional research on word-pair metaphor detection utilizes machine learning techniques based on linguistic or external knowledge related to metaphor, such as abstractness (Turney et al., 2011), imageability (Tsvetkov et al., 2014), visibility (Shutova et al., 2016) or property norm (Bulat et al., 2017). After that, some studies exploit neural networks modeling the similarity between words in a word pair (Rei

et al., 2017) or the concreteness constructed from images (Su et al., 2020a) to detect metaphors. In addition, Shutova et al. (2017) perform distributional clustering methods to identify metaphors. Recently, some researchers have utilized well-founded cognitive theory, conceptual metaphor theory (CMT) (Lakoff and Johnson, 1980), to benefit this task. Ge et al. (2022) design a method to generate explainable source and target domains to help identification of metaphors. Tian et al. (2023) further propose an attribute Siamese network to capture similar attributes between source and target concepts for metaphor detection.

Token-Level Metaphor Detection Inspired by a theory, selectional preference violation (SPV) (Wilks, 1975, 1978), which indicates that a metaphor contains violations between its context and its frequent usage of contexts, early work (Shutova et al., 2010; Li et al., 2013; Mao et al., 2018) develops statistical methods to capture co-occurrence information between words and their contexts based on various corpora and knowledge bases for token-level metaphor detection.

After that, some studies employ RNN-based models to extract contextual representations to detect token-level metaphors (Gao et al., 2018; Mao et al., 2019; Le et al., 2020). As these methods primarily rely on static word embeddings, they struggle to capture the intricate contextual meaning implied by metaphors. In contrast, pre-trained models (Devlin et al., 2019; Liu et al., 2019) can provide dynamic contextualized word embeddings and become popular backbones (Su et al., 2020b; Li et al., 2020) for current studies in metaphor detection. Among them, some studies employ the multitask framework to learn shared embeddings from other tasks related to metaphor (Mao and Li, 2021; Li et al., 2020; Zhang and Liu, 2023; Badathala et al., 2023). Other studies utilize external resources to enhance performance, such as multiword expressions (Rohanian et al., 2020) and word definitions (Su et al., 2021). In addition, some works employ data augmentation methods to expand limited metaphor datasets (Lin et al., 2021; Feng and Ma, 2022).

Recently, some studies (Choi et al., 2021; Zhang and Liu, 2022; Wang et al., 2023; Li et al., 2023) have developed methods to model the elements and their relations in metaphor identification theories, including SPV and metaphor identification procedure (MIP) (Pragglejaz Group, 2007), achieving promising results.

Computational Work for Both Metaphor Detection and Explanation Metaphor explanation aims to interpret the implicit meaning conveyed by metaphorical expressions, using paraphrased expressions (Shutova, 2010) or conceptual domains (Wachowiak and Gromann, 2023) as explanatory information. Previous works tackling both metaphor detection and explanation (Li et al., 2013; Mao et al., 2018) mainly employ sequential methods to detect metaphors and then explain them based on contextual frequency clues guided by SPV. In fact, metaphor detection and explanation are inherently associated, and intermediate explainable information can benefit metaphor detection. To bridge the gap between metaphor detection and explanation, Ge et al. (2022) propose a model to generate source and target domain for better word-pair metaphor detection guided by a more explainable theory CMT. However, their approach takes word pairs as inputs and relies on the simplified word-pair structure for domain mining. In contrast, token-level metaphor detection is a more general but challenging task involving complex sentence context. To mitigate this gap, we aim to advance token-level metaphor research by upgrading word-pair metaphor detection with explainable sentence-level conceptual domain modeling based on CMT.

3 Problem Definition

Formally, $\mathcal{D}_{tr} = \{(s^k, w_v^k, b^k)\}_{k=1}^{N_{tr}}$ is the training dataset with N_{tr} instances, where s is a sentence, w_v is a *focus verb* within s, and b is the label (metaphorical or literal) for w_v. $\mathcal{D}_{te} = \{(s^k, w_v^k, b^k)\}_{k=1}^{N_{te}}$ is the test dataset with N_{te} instances. The goal of token-level verb metaphor detection is to predict the label of the focus verb w_v in $\mathcal{D}_{te}$ by training a model with $\mathcal{D}_{tr}$.

4 Method

We propose an explainable **W**ord **P**air based **D**omain **M**ining (WPDM) method for token-level verb metaphor detection. Figure 2 illustrates the overview of our method, which contains four components: (1) *Literal Example Sentences Construction*, which collects example sentences for the literal meaning of the focus verb; (2) *Semantic Role Mapping*, which organizes words labeled with the same semantic role into a semantic group in the input sentence and literal example sentences, respectively; (3) *Conceptual Domain Mining*, which

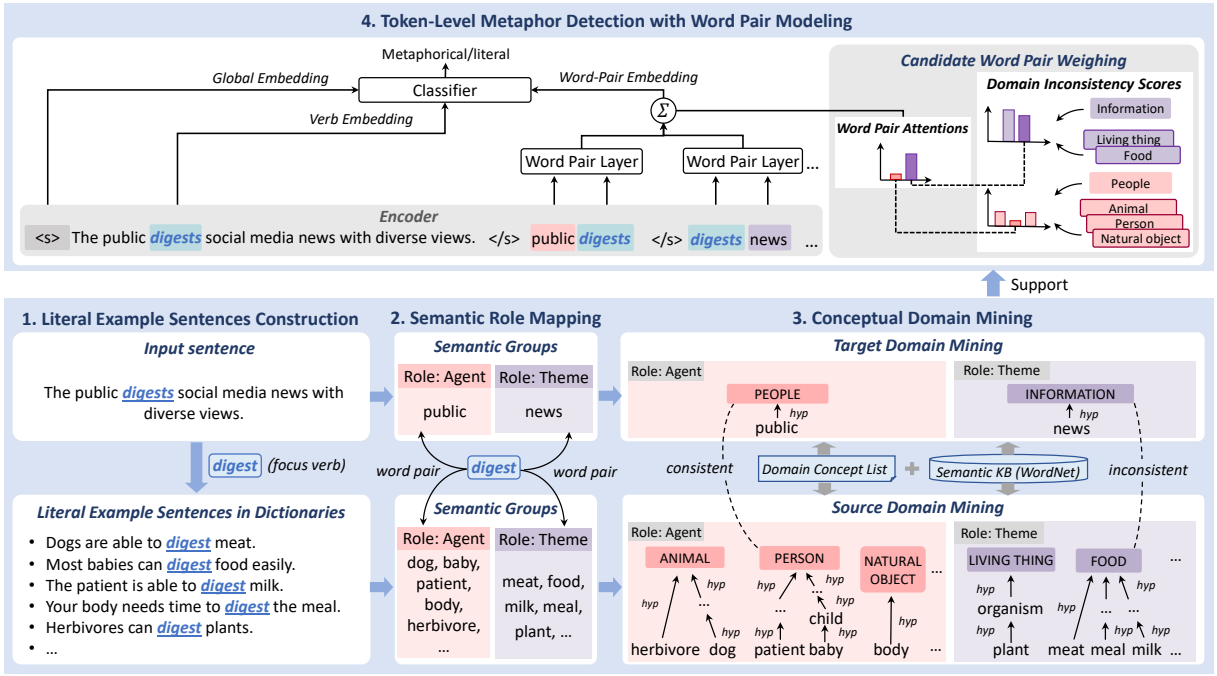

Figure 2: Overview of our method for token-level metaphor detection. (Here *hypernym* is abbreviated as *hyp*.)

employs a domain mining algorithm on words in each semantic group to mine the conceptual source and target domains based on cognitive, commonsense, and lexical resources; and (4) *Token-Level Metaphor Detection with Word Pair Modeling*, which evaluates the importance of word pairs in the sentence based on the inconsistency between conceptual source and target domains, and then gives more attention to the important core word pair during metaphor detection training.

4.1 Literal Example Sentences Construction

To obtain the context for the focus verb w_v using its literal meaning and establish a solid foundation for the following domain mining, we construct a literal example sentence set $S_e = \{s_e^k\}_{k=1}^{|S_e|}$ with $|S_e|$ sentences by gathering all the example sentences associated with the first sense of the focus verb w_v in dictionaries.[1]

4.2 Semantic Role Mapping

The semantic role of a verb, such as "agent" and "theme", describes the underlying relation between an argument (a phrase/word) and the verb in the sentence (Schuler, 2005). The argument labeled with a semantic role, which is closely related to the focus verb in semantics, can form a word pair with the focus verb to convey the implicit metaphorical meaning. To find these word pairs, we utilize VerbNet Parser to annotate the arguments (e.g. "public" in Figure 2) and their semantic roles associated with the focus verb w_v in both the input sentence s_{in} and the literal example sentences S_e. We then select the overlapping roles in s_{in} and S_e to construct the semantic role set $R = \{r_k\}_{k=1}^{|R|}$. Arguments labeled with the same semantic role in s_{in}

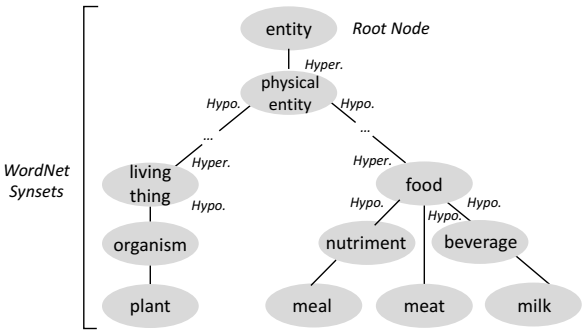

Figure 3: Illustration of the hierarchical structure of synsets in WordNet. A hypernym of a synset represents a broader or more general semantic field than the synset itself, while a hyponym of a synset conveys a more specific meaning. For example, "entity" is a hypernym of "physical entity" and "organism" is a hyponym of "living thing". *Hyper.* denotes hypernym and *hypo.* denotes hyponym.

[1]We use the first sense of a word in a dictionary to capture its literal usage, for the reason that dictionaries typically list word senses chronologically by starting with the original meaning (see https://www.merriam-webster.com/help/explanatory-notes/dict-definitions and https://www.oxfordlearnersdictionaries.com/faq).

construct a semantic group. We can get a collection of semantic groups $\mathcal{A}_{in} = \{A^i_{in}\}^{|R|}_{i=1}$ for s_{in}, where the set of the arguments labeled with the semantic role r_i is denoted as $A^i_{in} = \{(a^i_{in})_k\}^{|A^i_{in}|}_{k=1}$. Similarly, we can obtain a collection of semantic groups $\mathcal{A}_e = \{A^i_e\}^{|R|}_{i=1}$ for S_e, where the semantic group containing arguments labeled with r_i is denoted as $A^i_e = \{(a^i_e)_k\}^{|A^i_e|}_{k=1}$. The semantic groups A^i_{in} and A^i_e are mapped using semantic role r_i for the subsequent target and source domain mining.

4.3 Conceptual Domain Mining

Humans can formulate the taxonomy of domains from various perspectives, such as cognitive processes that provide the underlying framework for how humans think and categorize (Lakoff and Johnson, 1980), commonsense knowledge that offers categories of general facts and relationships (Speer et al., 2017), and lexical resources that reflect the linguistic aspect of concept organization (Gelman et al., 1989). To mine the conceptual target and source domains for the aforementioned mapped semantic groups, we develop a conceptual domain mining approach based on knowledge at cognitive, commonsense and lexical levels.

Domain Concept List We first construct a domain concept list that represents the taxonomy of domains established on human cognitive processes and commonsense knowledge, using two resources: Master Metaphor List (MML) (Lakoff et al., 1991), constructed by cognitive linguists, containing paired conceptual source and target domains of metaphorical understanding in cognitive processes, and OpenCyc (Lenat et al., 1985), a knowledge base consisting of large-scale commonsense concepts and relations. The combination of domain concepts from these resources constitutes the domain concept list, which is shown in Appendix A.

Domain Mining Algorithm A large lexical semantic database WordNet (Miller, 1995) organizes words into hierarchical structures through synsets and conceptual relations (hypernym and hyponym). Figure 3 shows the hierarchical structure of synsets in WordNet. Given the richness and transitivity of hypernym relations, WordNet is a valuable resource to identify candidate conceptual domains.

For a semantic group $A = \{a_i\}^{|A|}_{i=1}$, we regard the hypernyms along the hypernym path from a_i to the root node in WordNet as the candidate conceptual domains of a_i. Our goal is to construct a domain set $D = \{d_j\}^{|D|}_{j=1}$ and divide A into $|D|$ subsets $\hat{\mathcal{A}} = \{\hat{A}_j\}^{|D|}_{j=1}$, where every argument in the j-th subset $\hat{A}_j$ belongs to the j-th domain d_j. To assess how abstract a domain (hypernym) is relative to an argument, we introduce the abstraction level l, calculated by the relative distance between them in the hypernym path (e.g. the abstraction level of domain *food* relative to the argument *meal* is 2 in Figure 3). $(\theta_{abs})_j$ represents the average abstraction level of d_j for the arguments in $\hat{A}_j$, which is calculated by

$$(\theta_{abs})_j = \frac{1}{|\hat{A}_j|} \sum_{i=1}^{|\hat{A}_j|} l^i_j, \tag{1}$$

where l^i_j is the abstraction level of d_j relative to the i-th argument in A_j. The domains in D should strike a balance by capturing the broad meanings shared among the arguments in A without being overly abstract. For example, in Figure 3, *food* and *living thing* are more appropriate domains for the arguments (i.e. *plan*, *meal*, *meat*, and *milk*) than the excessively abstract domain *physical entity*. To evaluate how well D maintains this balance, we design the domain granularity metric θ_{dg}, which is calculated by

$$\theta_{dg} = \frac{1}{|D|} \sum_{k=1}^{|D|} \frac{|\hat{A}_k|}{(\theta_{abs})_k}. \tag{2}$$

Algorithm 1 Conceptual Domain Mining

Input: $A = \{a_i\}^{|A|}_{i=1}$: a semantic group with $|A|$ arguments.
Require: (1) L_d: the domain concept list; (2) $\mathrm{Comb}(A, k)$: a function to generate the combinations of k non-overlapping groups from the elements in the set A; (3) WordNet.
Output: $D = \{d_j\}^{|D|}_{j=1}$: the conceptual domain set.

for all $i \leftarrow 1$ **to** $|A|$ **do**
  for all $\hat{\mathcal{A}} = [\hat{A}_1, \ldots, \hat{A}_i] \in \mathrm{Comb}(A, i)$ **do**
    for all $\hat{A}_j \in \hat{\mathcal{A}}$ **do**
      if $\hat{A}_j$ contains only one argument $\hat{a}$ **then**
        Traverse h in hypernym path P of $\hat{a}$
        if $\exists$ ($h \in P$ and $h \in L_d$) **then**
          Obtain h as the domain for $\hat{A}_j$
        else
          Obtain $P[0]$ as the domain for $\hat{A}_j$
      else
        Obtain the least common ancestor hypernym of all arguments in $\hat{A}_j$ as the domain for $\hat{A}_j$
      Compute $(\theta_{abs})_j$ ▷ **Eq. (1)**
    Compute θ_{dg} ▷ **Eq. (2)**
  Obtain the domain set with the maximum θ_{dg} as D

We develop a conceptual domain mining algorithm to find all possible collections of subsets $\hat{A}$ for arguments in A and their related conceptual domains D based on the domain concept list and WordNet. We obtain the ones with the highest domain granularity metric θ_{dg} as our final results. Algorithm 1 shows the pseudocode of this algorithm.

We apply Algorithm 1 on every $A^i_{in} \in \mathcal{A}_{in}$ and $A^i_e \in \mathcal{A}_e$ to obtain the conceptual target domain set D^i_{in} for A^i_{in} and the conceptual source domain set D^i_e for A^i_e, respectively, where $i \in [1, |R|]$ is the index of semantic role.

4.4 Token-Level Metaphor Detection with Word Pair Modeling

According to conceptual metaphor theory (Lakoff and Johnson, 1980), metaphor implies the inconsistency between source and target domains. Inspired by this, we utilize the inconsistency scores between above conceptual source and target domains to mine core word pairs that convey primary metaphorical meaning within the input sentence, and give more attention to core word pairs during the training process for explainable metaphor detection.

Candidate Word Pair Weighing In the input sentence, each word labeled with a semantic role and the focus verb can constitute a candidate word pair. For the k-th such word w^k_p labeled with the semantic role r_k, we have mined its conceptual target domain d^k_t and its set of conceptual source domain D^k_s derived from the words labeled with the same semantic role r_k in literal example sentences, using our domain mining algorithm. According to CMT (Lakoff and Johnson, 1980), a metaphor implies the inconsistency between source and target domains. Thus, the candidate word pair showing high inconsistency between its target domain and all its source domains is likely to be the core word pair conveying primary metaphorical meaning. We assign an attention to each candidate word pair based on the minimum inconsistency score between its target domain and all source domains. Specifically, the inconsistency scores $\boldsymbol{\beta}_k$ between the target domain and all source domains for k-the candidate word pair and the attentions $\boldsymbol{\alpha}$ for all the candidate word pairs are calculated by

$$\boldsymbol{\beta}_k = (\mathrm{Sim}(d^k_t, D^k_s))^{-1} \in \mathbb{R}^{|D^k_s|}, \quad (3)$$

$$\hat{\beta}_k = \mathrm{Min}(\boldsymbol{\beta}_k), \quad (4)$$

$$\boldsymbol{\alpha} = \mathrm{Softmax}([\hat{\beta}_1, \ldots, \hat{\beta}_{n_p}]) \in \mathbb{R}^{n_p}, \quad (5)$$

where $\mathrm{Sim}(\cdot)$ calculates the path distance similarity in WordNet between d^k_t and every domain in D^k_s, $(\cdot)^{-1}$ denotes the operation of element-wise inverse, $\mathrm{Min}(\cdot)$ obtains the minimum score in a vector, $\mathrm{Softmax}(\cdot)$ is the softmax function, and n_p is the number of candidate word pairs.

Encoding To train our model from a good start of text embedding, we use the pre-trained model RoBERTa (Liu et al., 2019), as the text encoder. Given the input sentence s_{in} and candidate word pairs $P = \{p_k\}^{n_p}_{k=1}$, the input T is constructed by

$$T = \text{<s>}\ s_{in}\ \text{</s>}\ p_1\ \text{</s>}\ p_2 \ldots \text{</s>}\ p_{n_p}, \quad (6)$$

where <s> and </s> are the global and separation tokens, respectively. We divide T into tokens and feed them into RoBERTa to obtain contextual embeddings $H \in \mathbb{R}^{N \times d}$ and the contextual embedding of global token $\boldsymbol{h}_{\text{<s>}}$:

$$H = \mathrm{RoBERTa}(T) = [\boldsymbol{h}_1, \ldots, \boldsymbol{h}_N]^\top, \quad (7)$$

$$\boldsymbol{h}_{\text{<s>}} = \boldsymbol{h}_1 \in \mathbb{R}^d, \quad (8)$$

where d is the dimension of embedding, N is the number of tokens in T and $\boldsymbol{h}_i$ is the embedding for the i-th token in T. We also obtain the embedding of the focus verb $\boldsymbol{h}_v$ in the input sentence, and the embeddings of the focus verb $\boldsymbol{h}^k_v$ as well as the other target word $\boldsymbol{h}^k_w$ in the k-th candidate word pair. If a word is cut into tokens, its embedding is calculated by averaging its token embeddings.

Word Pair Layer The embedding for the k-th candidate word pair is computed as

$$\hat{\boldsymbol{h}}^k_{wp} = \mathrm{CrossAtt}(\boldsymbol{h}^k_v, \boldsymbol{h}^k_w) \in \mathbb{R}^d, \quad (9)$$

where $\mathrm{CrossAtt}(\cdot)$ denotes l_t Transformer blocks (Vaswani et al., 2017). In this module, $\boldsymbol{h}^k_v$ serves as the query vector and $\boldsymbol{h}^k_w$ serves as both key and value vectors. The aggregated word pair embedding $\boldsymbol{h}_{wp} \in \mathbb{R}^d$ is computed by

$$\boldsymbol{h}_{wp} = \sum_{k=1}^{n_p} \hat{\boldsymbol{h}}^k_{wp} \alpha_k, \quad (10)$$

where α_k is the k-th element in $\boldsymbol{\alpha}$.

Classification Finally, we feed the global embedding $\boldsymbol{h}_{\text{<s>}}$, verb embedding $\boldsymbol{h}_v$ and word pair embedding $\boldsymbol{h}_{wp}$ into a classifier, and adopt a cross-entropy loss function to compute the loss $\mathcal{L}_{class}$:

$$\hat{\boldsymbol{l}} = \mathrm{Softmax}(\mathrm{MLP}(\boldsymbol{h}_{\text{<s>}} \oplus \boldsymbol{h}_v \oplus \boldsymbol{h}_{wp})), \quad (11)$$

$$\mathcal{L}_{class} = -(\boldsymbol{l})^\top \log \hat{\boldsymbol{l}}, \quad (12)$$

Dataset	#Sent.	#Verb	%Met.	Avg. Len
MOH-X	647	647	48.69	8.0
LCC-Verb	2009	2009	42.61	29.1
TroFi	3,737	3,737	43.54	28.3
VUA-Verb$_{\text{train}}$	7,479	15,516	27.90	20.2
VUA-Verb$_{\text{validation}}$	1,541	1,724	26.91	25.0
VUA-Verb$_{\text{test}}$	2,694	5,873	29.98	18.6

Table 1: Statistics of datasets. **#Sent.** denotes the number of sentences. **#Verb** denotes the number of verbs that need to be detected. **%Met.** denotes the percentage of metaphor samples. **Avg. Len** denotes the average sentence length.

where $\oplus$ is the concatenation operation, $\mathrm{MLP}(\cdot)$ is a multilayer perceptron with hidden dimension d', $\hat{\boldsymbol{l}} \in \mathbb{R}^2$ is the predicted probability for all the labels, and $\boldsymbol{l} \in \mathbb{R}^2$ is the ground truth.

5 Experiments

5.1 Datasets

We constructed a new verb metaphor detection dataset **LCC-Verb** based on LCC (Mohler et al., 2016), which is one of the representative benchmark datasets for metaphor detection. The LCC-Verb construction process is shown in Appendix B. We also conducted experiments on three publicly available verb metaphor detection datasets, which are as follows: (1) **MOH-X** (Mohammad et al., 2016), containing 647 sentences where only a verb labeled as metaphorical or literal in each sentence; (2) **TroFi** (Birke and Sarkar, 2006), another verb metaphor detection dataset collected from Wall Street Journal Corpus; and (3) **VUA-Verb** (Leong et al., 2020), the largest publicly available dataset for verb metaphor detection drawn from VU Amsterdam Metaphor Corpus (VUA) (Steen et al., 2010). Table 1 shows the statistics of these datasets.

5.2 Baselines

We compare our method with several representative methods for token-level metaphor detection, which are as follows: (1) **RNN_ELMo** (Gao et al., 2018) is a BiLSTM-based method using ELMo embeddings to model metaphorical words in context; (2) **RNN_HG** and **RNN_MHCA** (Mao et al., 2019) make the first attempt to apply linguistic theories (MIP & SPV) on BiLSTM-based network design for metaphor detection; (3) **MUL_GCN** (Le et al., 2020) exploits a multi-task learning framework to transfer knowledge from word sense disambiguation to metaphor detection; (4) **MrBERT** (Song et al., 2021) regards metaphor detection as a relation classification task via extracting dependency relations and utilizing them for metaphor detection; (5) **MisNet** (Zhang and Liu, 2022) proposes a linguistics enhanced network inspired by MIP and SPV for metaphor detection; (6) **AdMul** (Zhang and Liu, 2023) proposes a multi-task learning framework to transfer knowledge from basic sense discrimination to metaphor detection via adversarial training; (7) **BasicBERT** (Li et al., 2023) proposes a method to mine the concise basic meaning of the word based on literal annotation from training set inspired by MIP.

5.3 Implementation Details

We use three English dictionaries to extract literal example sentences in our methods, including Longman Dictionary of Contemporary English[1], Oxford Advanced Learner's Dictionary[2], and Collins English Dictionary[3]. We use F1 score and accuracy for evaluation. Following the convention of previous studies (Zhang and Liu, 2022, 2023), we take the model that achieves the best F1 score on the validation set to test on VUA-Verb testing dataset, and we calculate the average over the best F1 scores and corresponding accuracy scores in total 10 folds for MOH-X, TroFi and LCC-Verb datasets. We use RoBERTa$_{\text{base}}$ (Liu et al., 2019) as the encoder. Given MUL_GCN (Le et al., 2020) did not release their code, we reproduced their method according to their paper and tested it on our four datasets. For a fair comparison, we replaced ELMo in MUL_GCN, BERT$_{\text{base}}$ in MrBERT and DeBERTa$_{\text{base}}$ in AdMul with RoBERTa$_{\text{base}}$ in our experiments. All experiments are done on NVIDIA RTX 3090 GPUs. Other details are illustrated in Appendix C.[4]

5.4 Main Results

From the experimental results shown in Table 2, we can see that our proposed method outperforms previous state-of-the-art performances across most datasets, which verifies the effectiveness of our core word pair modeling based on conceptual domain mining for token-level metaphor detection. Methods utilizing pre-trained models, which incorporate extensive world knowledge through unsu-

[1]https://www.ldoceonline.com/

[2]https://www.oxfordlearnersdictionaries.com/

[3]https://www.collinsdictionary.com/dictionary/english

[4]Our code is available at https://github.com/TIAN-viola/WPDM.

Method	MOH-X		LCC-Verb		TroFi		VUA-Verb		Average	
	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc
RNN_ELMo (Gao et al., 2018)	75.6	77.2	73.9	79.3	71.1	74.6	69.7	81.4	72.6	78.1
RNN_HG (Mao et al., 2019)	79.7	79.7	71.7	79.3	72.2	74.9	70.8	82.1	73.6	79.0
RNN_MHCA (Mao et al., 2019)	80.0	79.8	76.7	80.9	72.4	75.2	70.5	81.8	74.9	79.4
MUL_GCN† (Le et al., 2020)	78.7	78.0	77.3	79.2	71.6	74.7	69.6	81.8	74.3	78.4
MrBERT† (Song et al., 2021)	82.9	83.5	81.5	85.6	72.9	76.0	74.4	84.9	77.9	82.5
MisNet† (Zhang and Liu, 2022)	83.4	83.6	80.6	82.3	71.9	73.6	**75.9**	**86.0**	78.0	81.4
AdMul† (Zhang and Liu, 2023)	81.2	81.3	82.8	85.0	72.5	74.2	74.6	85.2	77.8	81.4
BasicBERT† (Li et al., 2023)	79.9	79.2	76.5	79.6	69.4	69.5	72.8	83.0	74.7	77.8
Our WPDM†	**83.8***	**84.2***	**84.9****	**87.0****	**73.4***	**76.3**	74.1	84.4	**79.0***	**83.0***

Table 2: Comparison between our method and baselines. The **best results** are in bold font and the second-best results are underlined. † indicates that these methods use the same $\text{RoBERTa}_{\text{base}}$ as the encoder. **Average** denotes the average results on four datasets. The symbols $*$ and $**$ indicate that our method's results are statistically significantly different from those of the second-best methods, with $p \leq 0.05$ and $p \leq 0.01$, respectively.

Method	MOH-X		LCC-Verb		TroFi		VUA-Verb	
	F1	Acc	F1	Acc	F1	Acc	F1	Acc
Our WPDM	**83.8**	**84.2**	**84.9**	**87.0**	**73.4**	**76.3**	**74.1**	**84.4**
w/o *Candidate word pair weighing*	83.2	83.5	84.6	86.4	72.8	75.3	73.1	83.7
w/o *Word-pair embedding* ($\boldsymbol{h}_{wp}$)	83.0	83.1	84.0	86.3	72.5	75.1	72.3	83.6
w/o *Global embedding* ($\boldsymbol{h}_{<s>}$)	82.0	82.1	84.2	86.4	73.3	75.4	73.0	83.7
w/o *Verb embedding* ($\boldsymbol{h}_v$)	80.8	81.8	80.9	83.6	67.8	71.7	65.6	80.8

Table 3: Results of ablation study. The **best results** are in bold font.

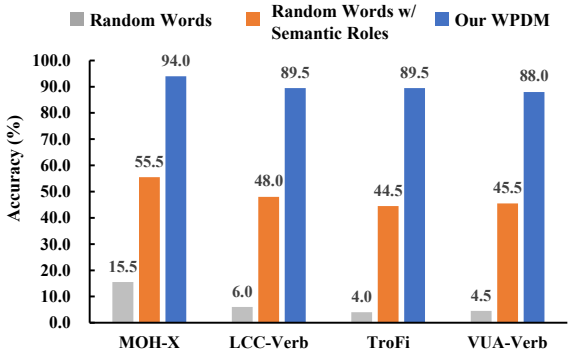

Figure 4: Evaluation results for core word pair selection.

Dataset	Target Domain	Source Domain
MOH-X	90.0	80.5
LCC-Verb	89.0	76.0
TroFi	82.5	82.0
VUA-Verb	83.5	82.5

Table 4: Results of human evaluation on domain mining.

pervised learning, perform better than earlier RNN-based models. Our proposed method achieves performance gains across most datasets compared to previous methods inspired by SPV and MIP, showing the superior advantage of the theory CMT utilized in our method over SPV and MIP.

5.5 Ablation Study

We conduct ablation study to evaluate the impact of components in our method. Table 3 shows the experimental results of the ablation study. Using average attentions to replace word pair attentions in candidate word pair weighing reduces the performance, verifying that candidate word pair weighing based on the inconsistency between source and target domains can help our model focus on the core word pair and benefit the explainable detection of metaphors. We further directly remove the word pair embedding from our method, resulting in greater performance drops across all datasets compared to only removing candidate word pair weighing, which verifies the effectiveness of word pair information in our method. When the global embedding is aborted, the performance decreases, thus demonstrating the importance of global context information for metaphor detection. In addition, as verb contains the central information for verb metaphor detection, the removal of verb embedding leads to the most significant performance declines across all the datasets.

Input Sentence (*focus verb*)	Core Word Pair	Source Domain	Target Domain
I *salute* your courage!	salute ***courage***	GENERAL_OFFICER	***SPIRIT***
He always *wears* a smile.	wears ***smile***	CHROMATIC_COLOR	***COMMUNICATION***
She *drowned* her trouble in alcohol.	drowned ***trouble***	PERSON	***DIFFICULTY***
The rules *relaxed* after the new director arrived.	***rule*** relaxed	PERSON	***IDEA***
The government *bowed* to the military pressure.	bowed ***pressure***	SOVEREIGN	***PHYSICAL_PHENOMENON***

Table 5: Case study for explainable results obtained by our method on MOH-X dataset. The paired source and target domains presented in this table are the paired domains that can calculate the highest inconsistency score among all the candidate source and target domain pairs in conceptual domain mining. The target word in the core word pair and its corresponding target domain are in ***bold font***.

5.6 Human Evaluation on Explainable Results

Our method can not only identify metaphorical verbs but also provide the core word pair along with conceptual source and target domains as explainable results. To evaluate these explainable results, we randomly sampled 100 metaphorical instances from MOH-X, LCC-Verb, TroFi and VUA-Verb's test dataset, respectively. We then invited two annotators to evaluate the core word pair as well as its source and target domains identified by our method using accuracy. The core word pair for each focus verb is the candidate word pair in the sentence with the highest word pair attention, and the corresponding paired domains used for the calculation of this attention are considered the source and target domains.

Core Word Pair Selection For the evaluation of core word pair selection, we compare our method with two baselines. The baseline *Random Words* randomly selects a word in each sentence to form a core word pair with focus verb. The other baseline *Random Words w/ Semantic Roles* randomly selects a word from the words labeled with semantic roles, which are related to the focus verb in the sentence, to form a core word pair with the focus verb. From the experimental results shown in Figure 4, we can see that our method outperforms baselines across all the datasets, verifying that our method can provide explainable intermediate clues in the form of the core word pair during metaphor detection, which is a capability lacking in previous methods.

Conceptual Domain Mining From the human evaluation results in Table 4, we can see that our method performs well on both target and source domain mining, with most metrics exceeding 80%, verifying the effectiveness of our conceptual domain mining algorithm. When comparing between the target and source domain mining, our method achieves better performance on mining target domains than source domains. This is because information related to source domains is often absent in the context of the focus verb, making it more challenging to identify source domains. In contrast, the information related to target domains is typically explicitly present in the context of the focus verb. Despite the difficulty in mining source domains, our method still achieves good performance across most datasets.

5.7 Case Study

Table 5 shows several cases for explainable results mined by our method on MOH-X dataset. For example, the word *drown* typically has a semantic relation of *theme* with nouns in the domain *PERSON*. However, in the third sentence of Table 5, its theme is *trouble*, which belongs to the target domain *DIFFICULTY*. The core word pair *drowned trouble* and the inconsistency between the source and target domains indicate a metaphorical use of the verb *drown* in this sentence. These cases further show the effectiveness of our method for explainable metaphor detection.

6 Conclusion

In this paper, we propose an explainable word pair based domain mining method for token-level metaphor detection. Inspired by conceptual metaphor theory, our method leverages semantic role mapping and conceptual source and target domain mining for core word pair modeling to facilitate explainable metaphor detection. Experimental results not only verify the effectiveness of our method for metaphor detection but also demonstrate its capability to determine the core word pair and conceptual domain information as explainable clues for the prediction of metaphors.

Limitations

Our work has some limitations. Currently, our method only focuses on detecting verb metaphors and providing explainable core word pair with corresponding source and target domains. Our future work needs to further explore other types of metaphors, such as noun metaphors and adjective metaphors. In addition, since our method utilizes external resources, our method may perform less effectively if verbs or context words cannot be found in these resources. Also, our method might exhibit performance decrease due to the fact that the external resources may be of lower quality in other languages.

Ethics Statement

As shown in previous studies (Navigli et al., 2023; Bartl et al., 2020), pre-trained language models exhibit biases in several respects, such as gender, ethnicity and race. Our method is fine-tuned on the pre-trained language model $\text{RoBERTa}_{\text{base}}$ and thus may inherit these biases.

Acknowledgments

This work is supported in part by the National Natural Science Foundation of China under Grants #72293575 and #62206287. We thank the anonymous reviewers for the valuable comments.

References

Naveen Badathala, Abisek Rajakumar Kalarani, Tejpalsingh Siledar, and Pushpak Bhattacharyya. 2023. A match made in heaven: A multi-task framework for hyperbole and metaphor detection. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 388–401.

Marion Bartl, Malvina Nissim, and Albert Gatt. 2020. Unmasking contextual stereotypes: Measuring and mitigating BERT's gender bias. In *Proceedings of the Workshop on Gender Bias in Natural Language Processing*, pages 1–16.

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

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. In *Proceedings of the Annual Conference on Neural Information Processing Systems*, pages 1877–1901.

Luana Bulat, Stephen Clark, and Ekaterina Shutova. 2017. Modelling metaphor with attribute-based semantics. In *Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics*, pages 523–528.

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 Conference of the North American Chapter of the Association for Computational Linguistics*, pages 1763–1773.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

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 Conference of the North American Chapter of the Association for Computational Linguistics*, pages 4171–4186.

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.

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

Mengshi Ge, Rui Mao, and Erik Cambria. 2022. Explainable metaphor identification inspired by conceptual metaphor theory. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 10681–10689.

Susan A. Gelman, Sharon A. Wilcox, and Eve V. Clark. 1989. Conceptual and lexical hierarchies in young children. *Cognitive Development*, 4(4):309–326.

George Lakoff, Jane Espenson, and Alan Schwartz. 1991. The master metaphor list. Technical report, University of California at Berkeley.

George Lakoff and Mark Johnson. 1980. *Metaphors we live by*. University of Chicago Press.

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

Doug Lenat, Mayank Prakash, and Mary Shepherd. 1985. CYC: Using common sense knowledge to overcome brittleness and knowledge acquisition bottlenecks. *AI magazine*, 6(4):65–85.

Chee Wee Leong, Beata Beigman Klebanov, Chris Hamill, Egon Stemle, Rutuja Ubale, and Xianyang Chen. 2020. A report on the 2020 VUA and TOEFL metaphor detection shared task. In *Proceedings of the Workshop on Figurative Language Processing*, pages 18–29.

Hongsong Li, Kenny Q Zhu, and Haixun Wang. 2013. Data-driven metaphor recognition and explanation. *Transactions of the Association for Computational Linguistics*, 1:379–390.

Shuqun Li, Jingjie Zeng, Jinhui Zhang, Tao Peng, Liang Yang, and Hongfei Lin. 2020. ALBERT-BiLSTM for sequential metaphor detection. In *Proceedings of the Workshop on Figurative Language Processing*, pages 110–115.

Yucheng Li, Shun Wang, Chenghua Lin, and Frank Guerin. 2023. Metaphor detection via explicit basic meanings modelling. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 91–100.

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 Conference on Empirical Methods in Natural Language Processing*, pages 3888–3898.

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. *Computing Research Repository*, arXiv:1907.11692.

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. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 13534–13542.

Rui Mao, Chenghua Lin, and Frank Guerin. 2018. Word embedding and WordNet based metaphor identification and interpretation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 1222–1231.

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

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

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

Michael Mohler, Mary Brunson, Bryan Rink, and Marc Tomlinson. 2016. Introducing the LCC metaphor datasets. In *Proceedings of the International Conference on Language Resources and Evaluation*, pages 4221–4227.

Roberto Navigli, Simone Conia, and Björn Ross. 2023. Biases in large language models: Origins, inventory, and discussion. *Journal of Data and Information Quality*, 15(2):1–21.

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

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

Omid Rohanian, Marek Rei, Shiva Taslimipoor, and Le An Ha. 2020. Verbal multiword expressions for identification of metaphor. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 2890–2895.

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

Ekaterina Shutova. 2010. Automatic metaphor interpretation as a paraphrasing task. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 1029–1037.

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

Ekaterina Shutova, Lin Sun, Elkin Darío Gutiérrez, Patricia Lichtenstein, and Srini Narayanan. 2017. Multilingual metaphor processing: Experiments with semi-supervised and unsupervised learning. *Computational Linguistics*, 43(1):71–123.

Ekaterina Shutova, Lin Sun, and Anna Korhonen. 2010. Metaphor identification using verb and noun clustering. In *Proceedings of the International Conference on Computational Linguistics*, pages 1002–1010.

Wei Song, Shuhui Zhou, Ruiji Fu, Ting Liu, and Lizhen Liu. 2021. Verb metaphor detection via contextual relation learning. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 4240–4251.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the AAAI Conference on Artificial Intelligence*, page 4444–4451.

Gerard J. Steen, Aletta G. Dorst, John B. Herrmann, Anna A. Kaal, Tina Krennmayr, and Trijntje Pasma. 2010. *A method for linguistic metaphor identification: From MIP to MIPVU*. John Benjamins, Amsterdam.

Chang Su, Weijie Chen, Ze Fu, and Yijiang Chen. 2020a. Multimodal metaphor detection based on distinguishing concreteness. *Neurocomputing*, 429:166–173.

Chang Su, Kechun Wu, and Yijiang Chen. 2021. Enhanced metaphor detection via incorporation of external knowledge based on linguistic theories. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 1280–1287.

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

Weiwei Sun, Shuyu Guo, Shuo Zhang, Pengjie Ren, Zhumin Chen, Maarten de Rijke, and Zhaochun Ren. 2023. Metaphorical user simulators for evaluating task-oriented dialogue systems. *ACM Transactions on Information Systems*, 42(1):1–29.

Yuan Tian, Nan Xu, Wenji Mao, and Daniel Zeng. 2023. Modeling conceptual attribute likeness and domain inconsistency for metaphor detection. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 7736–7752.

Yulia Tsvetkov, Leonid Boytsov, Anatole Gershman, Eric Nyberg, and Chris Dyer. 2014. Metaphor detection with cross-lingual model transfer. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 248–258.

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

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the Conference on Neural Information Processing Systems*, pages 1–11.

Lennart Wachowiak and Dagmar Gromann. 2023. Does GPT-3 grasp metaphors? identifying metaphor mappings with generative language models. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 1018–1032.

Shun Wang, Yucheng Li, Chenghua Lin, Loic Barrault, and Frank Guerin. 2023. Metaphor detection with effective context denoising. In *Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics*, pages 1404–1409.

Semantic Role List			
Asset	Instrument	Time	Agent
Medium	Location	Destination	Causer
Extent	Theme	Patient	State
Utterance	Stimulus	Recipient	Material
Attribute	Goal	Location	Source
Path	Pivot	Experiencer	Trajectory
Value	Manner	Circumstance	Product
Beneficiary	Topic	Result	

Table 6: Semantic roles parsed by VerbNet Semantic Parser.

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.

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

Shenglong Zhang and Ying Liu. 2023. Adversarial multi-task learning for end-to-end metaphor detection. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1483–1497.

A More Details on Method

A.1 Semantic Role Mapping

We use a publicly available tool[1] as the VerbNet Parser in our method. The list of semantic roles that we use is shown in Table 6. We split arguments containing conjunctions like "or" or "and" into separate arguments using these conjunctions and then perform lemmatization on them.

If the VerbNet Semantic Parser fails to label the semantic roles, we use Stanford dependency parser[2] as a substitute. Specifically, we use dependency relations, including *nsubj*, *nmod:agent*, *obl:agent*, *nmod:agent* and *agent*, to label the semantic role *Agent*, and dependency relations, including *obj*, *dobj*, *nsubjpass*, *nsubj:xsubj*, *nsubj:pass* and *nsubj:passdir*, to label the semantic role *Theme*.

[1]https://github.com/jgung/verbnet-parser
[2]https://stanfordnlp.github.io/CoreNLP/

Domain Concepts in Master Metaphor List (Lakoff et al., 1991) (Cognitive Level)					
BODY	IDEA	INJURY	CLOTH	ARGUMENT	CONTAINER
LOVE	HOPE	SOCIETY	BATTERY	LIGHT	ANGER
ABILITY	PROBLEM	DEATH	EMOTION	FIRE	WATER
HARM	RESOURCE	COMMODITY	WAR	FIGHT	CHILD
CAREER	BURDEN	RACE	WORD	INFORMATION	MONEY
JOURNEY	SCALE	FLUID	FAILURE	THEORY	CHANGE
OBLIGATION	PEOPLE	COMPETITION	LIQUID	OBJECT	RESPONSIBILITY
IMPORTANCE	FOOD	PRECEDENCE	MACHINE	MOTION	TIME
LIFE	BELIEF	PATH	WEAPON		
Domain Concepts in OpenCyc (Lenat et al., 1985) (Commonsense Level)					
GOAL	STATE	MOVEMENT	VEHICLE	HUMAN ACTIVITY	SOLAR SYSTEM
LOGIC	ARTIFACT	SHOPPING	COMMUNICATION	MATERIAL	HUMAN BEING
SOCIAL ACTIVITY	HUMAN	MATH	DEVICE	SPACE	POLITICS
PHYSIOLOGY	COMMERCE	TRANSPORTATION	FORM	SOFTWARE	ENTERTAINMENT
INDIVIDUAL	AGENT	PROFESSION	ACTION	CULTURE	EARTH
LIVING THING	ECOLOGY	BUILDING	SPORT	ANIMAL	LOGISTICS
PRODUCT	WEATHER	CHEMISTRY	BEHAVIOR	BUSINESS	OCCUPATION
LITERATURE	ORGANIZATION	PLAN	SOCIAL	TRAVEL	LANGUAGE
ASTRONOMY	WORK	RELATION	PLANT	LIFE	BELIEF
PATH	WEAPON	TIME			

Table 7: Domain concept list constructed from domains in Master Metaphor List and OpenCyc.

Notation	Value				Description
	MOH	LCC-Verb	TroFi	VUA-Verb	
N	160	200	200	200	maximum length of text tokens
$lr_{exc.\ encoder}$	$3e^{-3}$	$3e^{-4}$	$3e^{-4}$	$3e^{-4}$	learning rate of components except the text encoder
$lr_{encoder}$	$3e^{-5}$	$3e^{-5}$	$3e^{-5}$	$3e^{-5}$	learning rate of the text encoder
bs	32	32	32	512	batch size
l_{mlp}	2	3	3	3	the number of layers in $\mathrm{MLP}(\cdot)$

Table 8: Hyper-parameter values in our proposed method.

A.2 Conceptual Domain Mining

The domain concept list is shown in Table 7. When constructing the domain concept list, we filter out overly abstract domains (e.g., THING). We also exclude overly abstract hypernyms in WordNet, including *physical_entity*, *abstraction*, and *entity*. When determining the hypernym path of a word w in our conceptual domain mining algorithm, if the word w can be annotated with a name entity label (i.e. person, location, organization, money, number, ordinal, percent, date, time, duration), we obtain the connection of this label and its hypernym path in WordNet as the hypernym path for w; Otherwise, we directly obtain the hypernym path of w in WordNet.

In our method, we use the function $\mathrm{path\_similarity}(\cdot)$ in NLTK python packages[1] as the function $\mathrm{Sim}(\cdot)$ in Eq. (3) to calculate the path similarity of two words in WordNet. $\mathrm{path\_similarity}(\cdot)$ returns a score denoting the similarity between two word senses, based on the shortest path that connects the senses in WordNet. The score is in the range of 0 to 1.

B Dataset Construction

LCC-Verb LCC (Mohler et al., 2016) is a metaphor detection dataset with metaphoricity ratings of 0 to 3. We extracted verb instances to create LCC-Verb, classifying those labeled 0 as literal and those labeled 2 or 3 as metaphorical. We discarded instances labeled 1, which are possible metaphors.

C Implementation Details

We rerun the released code of baselines on our newly constructed LCC-Verb dataset. As BasicBERT has never been tested on the datasets that we use, we rerun their code[2] on four datasets in our

[1] https://www.nltk.org/howto/wordnet.html

[2] https://github.com/liyucheng09/BasicBERT/tree/master

Method	Template prompt
Standard zero-shot	Decide whether the word "[verb]" in the sentence "[sentence]" is used metaphorically. Give me an answer selected from "yes" or "no".
Few-shot	Q: Decide whether the word "[verb_example]" in the sentence "[sentence_example]" is used metaphorically. A: [answer_example] Q: Decide whether the word "[verb]" in the sentence "[sentence]" is used metaphorically.

Table 9: Prompt design for zero-shot and few-shot prompting strategies in ChatGPT experimentation. *[sentence]* represents the input slot for the sentence in an instance from the test dataset, and *[verb]* represents the input slot for the focus verb within this sentence. Similarly, *[sentence_example]* denotes the input slot for the sentence in an instance from the training dataset, while *[verb_example]* represents the input slot for the focus verb within this sentence. *[answer_example]* serves as the input slot for the answer with "yes" indicating a metaphorical instance and "no" indicating a literal instance.

Model	Prompting Strategy	MOH-X		LCC-Verb		TroFi		VUA-Verb	
		F1	Acc	F1	Acc	F1	Acc	F1	Acc
GPT-3.5	Standard Zero-Shot	64.4	67.3	42.6	50.4	50.1	56.0	50.5	69.3
	5-Shot	70.1	72.5	48.0	52.8	57.6	60.0	53.6	69.8
GPT-4	Standard Zero-Shot	72.6	77.0	66.4	66.0	67.3	67.5	66.9	75.5
	5-Shot	75.3	79.0	69.7	68.0	68.2	68.3	67.6	75.8
Our WPDM	-	**83.8**	**84.2**	**84.9**	**87.0**	**73.4**	**76.3**	**74.1**	**84.4**

Table 10: Comparison between our method and GhatGPT. The **best results** are in bold font.

experiments. We also rerun the code of MisNet[1] and AdMul[2] on LCC-Verb in our experiments. We use AdamW[3] as our optimizer with a weight decay of 0.01. The dropout rate is 0.5. The dimension of the text embedding d is 768. The hidden dimension of MLP is 2304. The layer of Transformer blocks is 2. We train our method for 15 epochs. Table 8 shows other hyper-parameter values. We utilize Stanford CoreNLP[4] for named entity tagging in our method.

The average Cohen's kappa coefficients κ (Cohen, 1960) of the inter-rater agreement in human evaluations of explainable results on MOH-X, LCC-Verb, TroFi, and VUA-Verb datasets are 0.75, 0.75, 0.70, and 0.73, respectively (note that $0.6 \leq \kappa \leq 0.8$ means substantial agreement).

D Experiments on ChatGPT

Implementation Details We also compare our method with ChatGPT. We use GPT-3.5 (gpt-3.5-turbo-0613)[5] and GPT-4 (gpt-4-0125-preview)[6] as baseline models. We explore two prompting strategies, including the standard zero-shot prompting and the standard few-shot prompting (Brown et al., 2020). Table 9 summarizes prompts used for the experiments on ChatGPT. We randomly sampled 200, 250, 400, and 600 instances from MOH-X, LCC-Verb, TroFi and test set of VUA-Verb, respectively, as the testing datasets for ChatGPT, and then we randomly sampled 5 instances from the remaining data as the few-shot instances. To cleanse the answer, we select the initial string of either "yes" or "no" in the response provided by ChatGPT, following the process of converting all uppercase characters to lowercase in the answer string.

Experimental Results From the experimental results in Table 10, we can see that GPT-4 performs better than GPT-3.5 for metaphor detection. Although the 5-shot prompting strategy leads to performance improvements for both GPT-3.5 and GPT-4 compared to the standard zero-shot prompting strategy, our method outperforms all ChatGPT baselines, which verifies the effectiveness of our method and indicates the difficulty and challenge inherent in the task of metaphor detection.

[1] https://github.com/SilasTHU/MisNet
[2] https://github.com/SilasTHU/AdMul
[3] https://pytorch.org/docs/stable/generated/torch.optim.AdamW.html
[4] https://stanfordnlp.github.io/CoreNLP/ner.html
[5] https://platform.openai.com/docs/models/gpt-3-5
[6] https://platform.openai.com/docs/models/gpt-4-and-gpt-4-turbo

Scientific Artifact	License
MOH-X	https://saifmohammad.com/WebPages/SentimentEmotionLabeledData.html
TroFi	https://www.cs.sfu.ca/~anoop/students/jbirke/LICENSE.html
Stanford CoreNLP	GNU General Public License (v2 or later)
WordNet	WordNet 3.0 license
VUA-Verb	CC BY-SA 3.0
ConceptNet	CC BY-SA 4.0
roberta-base	MIT license
ChatGPT	API license
LCC	Unspecified
OpenCyc	Apache-2.0
NLTK	Apache-2.0
VerbNet Parser	Apache-2.0

Table 11: Licenses of the scientific artifacts.

E Licenses of Scientific Artifacts

Table 11 shows the licenses of the scientific artifacts used in this paper.