Can ChatGPT's Performance be Improved on Metaphor Detection Tasks? Bootstrapping and Combining Tacit Knowledge

Anonymous ACL submission

Abstract

Metaphor detection, as a key task in the field of natural language processing, has received sustained academic attention in recent years. Current research focuses on the development of supervised metaphor detection systems, which usually require large-scale, high-quality labeled data support. With the rapid development of large-scale generative language models, e.g., ChatGPT, they have been widely used in multiple domains, including automatic summarization, sentiment analysis, and question and answer systems. However, it is worth noting that the use of ChatGPT for downstream metaphor detection tasks is often challenged with less-than-expected performance. Therefore, we propose a new method that aims to fully utilize the implicit knowledge of ChatGPT to support the task of detecting zero-shot verb metaphors. The method first uses ChatGPT to generate literal meaning collocations of verbs. For the text to be detected, subject-object pair of the target verbs in the text are parsed. Subsequently, these literal collocations and subject-object pair are mapped to the same set of topics, and the metaphors are finally identified through the analysis of entailment relations. The results show that the performance of ChatGPT in the verb metaphor detection task can be significantly improved by bootstrapping and integrating the implicit knowledge of ChatGPT.

1 Introduction

Metaphors are essentially mapping relationships between two different domains (Hesse, 1965; Lakoff and Johnson, 2008). According to Lakoff and Johnson (2008)'s theory of conceptual metaphors, linguistic metaphors derive from underlying conceptual metaphors that map a source concept (source domain) to another, more abstract, domain target concept (target domain). The goal of automatic metaphor detection is to model non-literal expressions (e.g., metaphors and metonymy) and generate corresponding metaphor annotations. Improving metaphor detection is important for improving many natural language processing (NLP) tasks, including information extraction (Tsvetkov et al., 2013), sentiment analysis (Cambria et al., 2017), and machine translation (Babieno et al., 2022).

Metaphor detection as an important part of the field of Natural Language Processing (NLP), has a variety of outstanding approaches emerge in recent years. In terms of supervised classification, Su et al. (2020) delved into the application of localized textual information, which is reduced to the position of the target word in a sentence segment. Meanwhile, Choi et al. (2021) was the first to introduce the Metaphor Identification Program (MIP) (Group, 2007) and (SPV) (Wilks et al., 2013) structures into a pre-training model. They also developed a multi-task-based gating mechanism in which lexical annotation was introduced as an auxiliary task. In addition, Zhang and Liu (2023) also proposed a multi-task learning approach that facilitates knowledge fusion between different tasks by means of adversarial learning.

Supervised methods mostly rely on carefully labeled datasets, and although they show excellent performance on the corresponding test sets, they perform poorly when generalized to different domains. In the field of unsupervised metaphor detection, Heintz et al. (2013) constructed a topic table based on the latent Dirichlet allocation (LDA) and aligned it to the source and target domains, respectively. While Shutova and Sun (2013) constructed a clustering map based on grammatical features of verbs, the metaphor detection system of Gandy et al. (2013) relied on lexical abstraction. Furthermore, Pramanick and Mitra (2018) calculated the abstraction levels of adjectives and nouns separately, along with the cosine distances between them, and subsequently employed the k-means algorithm for clustering. While Mao et al. (2018); Shutova et al. (2016) employed cosine similarity to determine whether the focal words belong to the same conceptual domain. Although the aforementioned approaches achieved a certain level of advancement, they frequently depended on intricate manual coding rules (Heintz et al., 2013; Shutova and Sun, 2013; Gandy et al., 2013) or cannot completely escape the reliance on manually labeled datasets (Mao et al., 2018; Shutova et al., 2016).

To address the above problems, this paper proposes a zero-shot metaphor detection method designed to bootstrap and integrate the implicit knowledge of ChatGPT. This method does not require the construction of cumbersome manual coding rules, nor does it rely on manually labeled data. First, we create a verb table that recorded each verb literal meaning collocation. Next, we introduce topical features that map the subject and object of the target verb to one or more topical categories. In the metaphor detection process, we first analyze the subjects and objects of the verbs to be detected in the input text and map them to topical categories as well. Finally, we make metaphor judgments based on Selectional Preference Violation (SPV) (Wilks et al., 2013). We tested it on the MOH-X and TroFi datasets, and the results show that by bootstrapping and integrating the implicit knowledge of a large language model, we can effectively improve its performance on the metaphor detection task.

In summary, the main contributions of this paper are summarized as follows:

1. We are the first to introduce ChatGPT to the task of metaphorical sequence annotation. Our method do not need to rely on tedious hand-coding rules or manually labeled data.

2. We used ChatGPT to generate a verb table that provides reference information about all literal meaning collocations for each verb.

3. We introduce topical features that act as additional semantic information to provide the method with richer background knowledge.

4. The experimental results show that by bootstrapping and integrating implicit knowledge from a large language model, the performance of ChatGPT on the metaphor detection task is significantly improved.

2 Related Work

The task of metaphor detection has been received a lot of attention in the field of natural language processing. Karov and Edelman (1998) used a word sense disambiguation (WSD) algorithm to cluster sentences with target words, and then made metaphor predictions based on the principle of distance between literal meanings of words. Shutova and Sun (2013) also drew on the idea of clustering, and it used the Gigaword corpus (Graff et al., 2003) with noun-related of verb-noun combinations (grammatical features) to cluster the 2000 common nouns of the BNC. In this approach, the words to be detected acquire knowledge information at a certain layer in the clustering map, i.e., the nouns at that layer are non-metaphorically related to the words to be detected.

Mao et al. (2018) presented an approximately unsupervised metaphor detection system. The system selects the best alternative to the target word by considering superlatives and synonyms in the context. When the cosine distance between the best alternative and the target word is greater than a specific threshold, it is detectd as a literal meaning. In addition, other studies Shutova et al. (2016); Pramanick and Mitra (2018) have considered the cosine distance, although Pramanick and Mitra (2018) did not use a priori labeled data to set the threshold, instead it adopted a feature construction approach using clustering for metaphorical judgments.

The studies in Turney et al. (2011); Gandy et al. (2013) explored the relationship between the abstraction degree of focus words and the expression of language metaphors. In Turney et al. (2011), the abstraction degrees of nouns, proper nouns, verbs and adverbs were first calculated, and then logistic regression was used to learn high-dimensional metaphoric features. In contrast, Gandy et al. (2013) used WordNet to generate n common collocations of the words to be detected and sorted these collocations according to the abstraction level. A metaphorical relationship word is detectd as a metaphor if it is not between the first k most concrete words. This idea is also reflected in the study of Krishnakumaran and Zhu (2007), which investigated three metaphorical relations, Subject-be-Object, Verb-Object and Adjective-Noun, and identified metaphors by determining whether the two focal words have a hyponymy relation.

Although the above methods have been effective to a certain extent, there are still problems such as complex parsing of metaphorical relationships, cumbersome construction of hand-coded knowledge, or over-reliance on manually labeled data. To overcome these challenges, this paper attempts to introduce generative language modeling into the metaphor detection task. The main function of generative language models is to generate natural language text, which can be used for conversing with humans or performing text generation tasks. These models perform self-supervised learning from large-scale textual data without relying on task-specific labeling or guidance.

In previous research, Wachowiak and Gromann (2023) introduced generative language modeling to the field of metaphor detection for the first time, albeit with only preliminary attempts. This study first provided input text and target domain information, and then utilized ChatGPT to predict source domain information and achieved a weighted accuracy of 60.22% on the combined dataset. Inspired by this research, this paper introduces ChatGPT to the task of metaphorical sequence annotation and achieves significant performance improvements by bootstrapping and combining the model's tacit knowledge.

3 Method

In this section, we present the zero-shot metaphor detection method in detail, dividing its core concepts into three parts: Defining Verb Metaphors, Topic Mapping, and Construction of Verb Lists. The last subsection elaborates on the specific implementation details of the proposed method.

3.1 Defining Verb Metaphors

Our study about verb metaphors is based on the theory of Selectional Preference Violation (SPV) (Wilks et al., 2013). As an important concept in linguistics, SPV reflects the relatedness and semantic compatibility between lexical units. For example, in the phrase "kill time", the verb "kill" is originally preferred to describe the behavior of animate objects, but here it modifies the inanimate "time", so there is a case of Selectional Preference Violation.

In previous studies, Shutova et al. (2012, 2016) usually categorized verb-metaphor relations into two main types, i.e., Subject-Verb (SV) pair and Verb-Direct Object (VO) pair. For example, in the sentence "He planted good ideas in their minds.", "ideas" is the direct object of the verb, and the verb "planted" forms a VO pair with "ideas". the subject of the target verb "planted" is "he", which forms an SV pair. To capture the metaphorical relations of verb pair more comprehensively, we considered both SV pair and VO pair. We consider the target verb to be non-metaphorical only if both sub-relations exhibit literal meaning relations.

In other studies, Krishnakumaran and Zhu (2007); Gandy et al. (2013) have also introduced Subject-be-Object (SbeO) relations. For example, in the sentence "Her love is a warm blanket on a cold night.", "love" is metaphorized as a warm blanket. In this structure, the verb "is" connects two focus words, "love" and "blanket". However, it should be noted that "is" as an auxiliary verb does not have an independent lexical meaning by itself; it needs to be combined with other verbs. Therefore, when judging the metaphor of SbeO relations, it is necessary to consider whether there is an entailment relationship between the subject or object. This is more similar to the Adjective-Noun (AN) relation pair discussed by Pramanick and Mitra (2018). Therefore, we categorize SbeO relations in the same category as AN pair, instead of including them among the verb metaphors studied.

3.2 Topic Mapping

Metaphorical relationships originated from conceptual mappings in different domains (Lakoff and Johnson, 2008). Inspired by it, we introduce the concept of topic, which can be viewed as broader and abstract concepts to correspond to domains in metaphors. Consider an example of a verb metaphor using the Oxford topic, the verb "guzzle" is often used with the subjects "baby" and the objects "milk". However, in the sentence "The car guzzled down the gasoline.", the subject and object of the target verb "guzzled" are "car" and "gasoline", respectively. This leads to the verb selective preference violation. In addition, since "bus" or "taxi" belongs to the same topic "Transport by car or lorry" as "car". Therefore, replacing the subject of the above example sentence with "bus" or "taxi" also constitutes a metaphorical expression.

We introduce three kinds of topics, namely Oxford topics, WordNet topics, and LDA topics. These three topic categories are set up in line with both the SPV (Wilks et al., 2013) and the abstractness principle defined in Turney et al. (2011); Gandy et al. (2013). The principle of abstraction holds that focus words under the same topic usually have similar or close levels of abstraction. For example, in the example in the Oxford topic, "Anger," "Fear," and "Happiness" all belong to the "People-Feelings" topical category, and these words have similar levels of abstraction. However, it is important to note that, since a single word may have more than one denotation, the word may correspond to more than one different Oxford topic.

The LDA topics were derived from a category list containing 60 topics constructed by Heintz et al. (2013). The method first used the LDA (Blei et al., 2003) model to capture a variety of candidate topics from WiKipedia. Then, based on the metaphorical information contained in the input corpus, the topics with high relevance to metaphorical relations were selected as the final metaphorical topics, and they were summarized into 60 different topic categories. The constructed topics would be categorized according to the order of similarity in WordNet from high to low for the central words.

Similar to the infix relation defined in Krishnakumaran and Zhu (2007), we introduce the set of superlatives and synonyms in WordNet (Kilgarriff, 2000) as a third topic (WordNet topic). In WordNet, superordinates are defined as semantically more general or abstract words, while synonyms denote words with similar or identical meanings that can provide complementary information. Since both superlatives and synonyms are considered, each central word in a WordNet topic contains all synonyms and superlatives compared to LDA topics that select one or more topics by similarity.

3.3 Construction of Verb Lists

Currently, supervised metaphor detection systems (Choi et al., 2021; Zhang and Liu, 2023) usually require large-scale labeled data for training to learn the generalized distribution of metaphors. However, this data labeling process is time-consuming and labor-intensive, thus limiting its feasibility in large-scale applications. Furthermore, when supervised models are applied to transfer learning, a sharp decrease in their performance in new domains is often observed (Wang et al., 2023). This phenomenon suggests the existence of a domain bias problem (i.e., a significant difference between the metaphor dataset and the actual metaphor application environment). In addition, the dynamic nature of metaphors is also a challenge (Shutova et al., 2013). Over time, old metaphors may gradually evolve into generic expressions, e.g., "email" initially denoted the transmission of messages over a physical distance, but with the popularity of sending emails over the Internet, it gradually evolved into the literal meaning of "sending and receiving emails". Therefore, models trained on traditional datasets (e.g., TroFi or MOH-X) may be difficult to adapt to the metaphorical usage contexts of real-world applications.

To address these challenges, we construct a verb collocation table. This verb list requires no additional training and can be used to establish a metaphorical reference standard appropriate to a particular need. As in the above example "Email me the report", we categorize the VO pair "Email me" as a literal relation to adapt to the current language usage. However, given that the main goal of this paper is to investigate the rationality of using verb lists in a zero-shot metaphor detection, we did not consider artificially customized verb lists.

Subject(Topic)	**Object(Topic)**
person (people)	Food or meals (Cooking and eating)
Children (Life stages)	Snacks (Cooking and eating)
Adults (Life stages)	Meat (Food)
diners (Cooking and eating)	Vegetables (Food)

Table 1: The subject and object of the verb "eat" are literally paired, with the corresponding Oxford topic category indicated in parentheses.

In this experiment, we generate literal or non-metaphorical collocations of verbs using GPT-3.5 Turbo (hereafter Turbo), a lightweight text generation model developed by OpenAI that can be adapted to a wide range of use cases through fine-tuning. First, we use the Turbo model to generate subject and object collocations for the target verbs. Then, SV and VO pairs are extracted separately by regular expressions and stored as a list. Noting that each target verb corresponds to two lists (i.e., the subject list and the object list), which do not correspond to each other. Next, we map the subject and object contents of the lists to one or more topics (see Section 3.2 for details), and the same topics for the same verb will be merged. Table 1 shows the Oxford topical information for the verb "eat". In the table, both "Children" and "Adult" belong to the topical category 'Life stages', so they are merged into the same category. Similarly, the object content of "Food and meals", "Snacks", "Meat" and "Vegetables" are categorized separately.

3.4 Method Implementation Details

In this section, we will delve into SVO-type verb metaphor relations, and the detailed details of the related algorithms can be found in Algorithm 1. First, we build a table of containing verbs D as described in Section 3.3. This verb table is in the form of a dictionary, where each particular verb is used as an indexing keyword, and the corresponding subject or object is stored in the form of a list, labeled as S_w and O_w, respectively. To perform metaphor detection, the input text needs to be processed first. Similar to the manipulation of verb lists, we will extract the subject and object in each input text.

In previous studies, researchers Wilks et al. (2013); Shutova et al. (2016); Gandy et al. (2013) usually used the Stanford Dependency Parser to extract SV and VO pairs of metaphorical relations, while another study Krishnakumaran and Zhu (2007) employed PCFG (Klein and Manning, 2003) for grammatical parsing. However, these approaches usually require the specification of complex rules to take into account complex grammatical structures such as inversions, implied subjects or objects, and subordinate clauses. Concretely, the Turbo model is used to generate the subject-verb-object structure of sentences. For each input sample n, we use regular expressions to parse the results generated by Turbo and store them as a list. If the generated SV or VO pair contain pronouns or named entities, we first obtain their basic meanings in the Oxford dictionary. For example, "it" corresponds to "used to refer to an animal or a thing that has already been mentioned or that is being talked about now". In this case, we usually choose the first 3 nouns (if they exist) as the center words of "it", such as "animal" and "thing".

Since the subjects and objects in the SV or VO pair output by the model are usually presented as phrases, we will select the first k nouns in the phrases as the center words of the subjects or objects and notate them as subj_nouns and obj_nouns, respectively. Then, depending on the lexical meaning of these center words, we map them to one or more topics, denoted as subj_topics and obj_topics, respectively. For example, in the sentence "He was detained on June 23, and for two weeks he was regularly assaulted by South African police", the subject of the sentence is "South African police". We extract the first k nouns as the center word, i.e., "police". According to the lexical meaning, we map "police" to the Oxford topic "Law and justice". Finally, we make metaphorical judgments based on the relationship between the parsed topics and the reference topics in the verb list.

4 Experiments

In this section, we detail the dataset used, the experimental steps, and perform an in-depth analysis of the results.

Dataset	Tokens	Sent.	%Met.
MOH-X	647	647	48.7%
TroFi	3,737	3,737	43.5%

Table 2: Statistical information on MOH-X and TroFi. "Tokens" denotes the total number of sentences, "Sent." denotes the number of samples, and "%Met" denotes the percentage of metaphorical samples.

4.1 Test Datasets

To evaluate our approach, we use the MOH-X (Birke and Sarkar, 2006) and TroFi (Charniak et al., 2000) datasets. The statistics of these two datasets are presented in Table 2.

MOH-X. The MOH dataset was originally created by Mohammad et al. (2016), who first extracted polysemous verb samples from WordNet, and then hired 10 annotators through the crowdsourcing platform CrowdFlower3 to metaphorically annotate the sentences. To ensure the annotation quality of the dataset, Mohammad et al. (2016) used the principle of 70% annotation consistency. Furthermore, they claimed that their sample contained only two categories, literal or metaphorical, which is consistent with our hypothesis. Here, we consider only the subset of verbs (i.e., MOH-X) in the MOH dataset processed according to Shutova et al. (2016). This subset excludes instances with pronouns or subordinate subjects or objects. The dataset ultimately contained 647 verb-noun combinations, of which 316 pairs are metaphorical and 331 pairs are literal. During data preprocessing, we use a specialized tool to extract the subject-verb-object relationship of each verb to be detected and removed samples that are incorrectly parsed or lacked subjects and objects. It is worth mentioning that the MOH-X dataset we used is not further divided into a training set and a test set, but is used as a whole for model testing and evaluation.

TroFi. The TroFi dataset (Birke and Sarkar, 2006),

Algorithm 1 Metaphor Detection

Require: D: Dictionary of verb forms
Require: S_w: List of literal or non-metaphorical subject topics for each target verb
Require: O_w: List of literal or non-metaphorical object topics for each target verb
Require: N: Input corpus containing sentences with target verbs
Require: w_n: Target verb in sentence n
Require: i_n: Index of the target verb in sentence n

```
for n in N do
    S_{w_n} ← D[w_n][0]                                  ▷ Retrieve subject topics
    O_{w_n} ← D[w_n][1]                                  ▷ Retrieve object topics
    Extract the subject and object from the sentence at index i_n.
    subj_nouns ← get_top_k_noun(subject)
    obj_nouns ← get_top_k_noun(object)
    subj_topics ← get_topics_from_oxford(subj_nouns)
    obj_topics ← get_topics_from_oxford(obj_nouns)
    if_sub_literal ← subj_topics ∈ S_{w_n}               ▷ Is subject literal?
    if_ob_literal ← obj_topics ∈ O_{w_n}                 ▷ Is object literal?
    if ¬(if_sub_literal ∧ if_ob_literal) then
        if_metaphor ← True                               ▷ Metaphor detected
    else
        if_metaphor ← False                              ▷ No metaphor
    end if
end for
```

derived from the Wall Street Journal corpus (Charniak et al., 2000), contains literal and metaphorical usage of 50 English verbs, totaling 3,717 samples, for the study of verb metaphors. Compared to the MOH-X dataset, the subject and object collocations with the target verbs in the TroFi dataset are more diverse, including pronouns, clauses, and named entities, which increases the complexity of metaphor detection. Consistent with our treatment of the MOH-X dataset, we extract subject-verb-object features for each sample in the TroFi dataset and excluded cases where parsing was wrong or where both subject and object were absent. It is worth noting that similar to the MOH-X dataset, the TroFi dataset is not further divided into training and testing sets.

4.2 Experimental Setup

Three different topics are considered in this experiment, including WordNet topics, LDA topics, and Oxford topics. For the WordNet topic, we use WordNet's built-in API to extract the superlatives and synonyms of the central noun, and then combine all of them into the WordNet topic set corresponding to the target verb. For the second topic, we use Wu-Palmer Similarity (WUPS) (Shet et al., 2012) to compute the similarity between the central noun and the 60 LDA subject terms. WUPS relies on lexical relations and hierarchical structures in the WordNet database. In the lexical relation network, it finds the Lowest Common Subsumer (LCS) of two words in WordNet. Then, the similarity is determined by calculating the path length between them and the LCS. The formula for similarity is usually shown below:

$$WUPS\left(w_1, w_2\right) = \frac{2 \cdot \text{depth}\left(LCS\left(w_1, w_2\right)\right)}{\text{depth}\left(w_1\right) + \text{depth}\left(w_2\right)},$$

where ω_1 and ω_2 represent the two words to be detected, LCS denotes their lowest common ancestor, and "depth" denotes the depth of the word in the WordNet hierarchy. For Oxford topics, we first access the Oxford lexicon for pronoun disambiguation and named entity conversion, and then convert the parsed central noun into one or more topic categories corresponding to the Oxford lexicon, if applicable, based on one or more lexical meanings of the parsed central noun. Since each subject or object in the target verb list usually contains multiple central nouns, the same topical transformation step needs to be performed for each central noun.

Concretely, we first parse the input text to extract the subject and object corresponding to the target

Models	TroFi				MOX-H			
	Acc.	**Pre.**	**Rec.**	**F1**	**Acc.**	**Pre.**	**Rec.**	**F1**
GPT-3.5 Turbo	58.7	11.4	64.2	19.3	60.1	20.0	91.3	32.8
WordNet_Topic	46.0	96.8	44.6	**61.0**	53.6	90.1	51.4	65.4
WordNet_Topic_k	46.2	95.9	44.5	60.6	54.1	88.6	51.7	65.3
LDA_Topic	45.9	91.4	44.2	59.6	51.2	94.0	50.0	65.3
LDA_Topic_k	44.5	96.9	43.9	60.4	52.2	92.9	50.3	65.3
Oxford_Topic	47.0	90.4	44.6	59.8	62.9	86.7	58.1	69.6
Oxford_Topic_k	45.8	93.7	44.2	60.1	61.2	93.3	56.1	**70.1**

Table 3: Performance comparison of TroFi and MOX-H datasets. The WordNet_Topic, LDA_Topic, and Oxford_Topic represent three different topics, respectively. The ones ending with "k" indicate that the first three nouns are extracted as the center nouns, while the ones without "k" indicate that one is extracted.

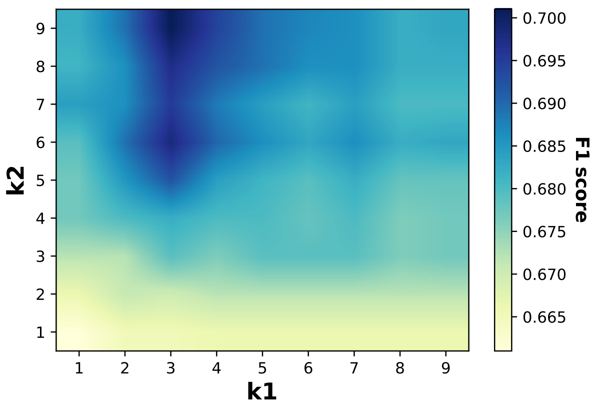

Figure 1: Effect of parameters $k1$, $k2$ on model performance, where k_1 represents the number of literal or non-metaphorical collocations selected from the verb list and k_2 denotes the number of topics that may be covered by the subject and object corresponding to the target verb.

verb (labeled as "none" if they do not exist). Since subject-object pair usually contain multiple nouns or proper nouns, we select the first k nouns as the subject content to be transformed by default, where k is a hyperparameter for the number of central nouns to be extracted. To balance the set size and metaphor detection accuracy when introducing the topic set, we also introduce two additional hyperparameters for control. Specifically, k_1 represents the number of literal or non-metaphorical collocations selected from the verb list, while k_2 denotes the number of topics that may be covered by the subjects and objects corresponding to the target verbs. Larger values of k_1 imply that the model's predictions cover more literal-meaning collocations of verbs, while larger values of k_2 indicate that more meanings of centered words are used in the metaphorical relations parsed in the text.

In the first experiment, we process different ways of extracting the central nouns of the subject or object in the input text, including the case of extracting 1 or 3 central nouns, which is achieved by adjusting the hyperparameter k. We chose the default k_1 and k_2 optimal combination approach for our experiments, and the specific types include WordNet_Topic, WordNet_Topic_k, LDA_Topic, LDA_Topic_k, Oxford_Topic, and Oxford_Topic_k, where k denotes the extraction of the first 3 nouns as the center nouns, while WordNet_Topic, LDA_Topic, and Oxford_Topic correspond to three different topics. It is worth noting that we use GPT-3.5 Turbo as the parsing tool when constructing the verb table. Therefore, we also conduct a controlled experiment to predict the results of the input corpus directly using GPT.

For the second experiment, we explore the effect of two hyperparameters, $k1$ and $k2$, on the model metaphor detection performance. For the experimental design, we used only Oxford topics. Considering the results of Experiment 1, we find that Oxford_Topic_k with three central nouns extracted performs better relative to Oxford_Topic with one central word extracted. In addition, when only one central noun is extracted, there are relatively fewer topic types (which depends on the number of different meanings of that central noun). Therefore, in this experiment, we fixed the hyperparameter of the central term to $k = 3$, while setting the value range of $k1$ and $k2$ between 0 and 9.

4.3 Results and Discussion

We use four common evaluation metrics, i.e., accuracy, precision, recall, and F1 score, to evaluate our approach.

For Experiment 1 (see the results in Table 3), the best performance is achieved on the entire TroFi dataset using the WordNet topic with an F1 score of 61.0%. And on the MOX dataset, the best performance is obtained using the Oxford topic, with an F1 score of 70.1%. For the hyperparameter k, we observe no significant performance difference between the two datasets by setting k to 1 or 3 when using WordNet topics or LDA topics. However, setting k to 3 slightly improves the performance when using the Oxford Dictionary topic. This may be due to the presence of polysemy in Oxford topics (i.e., different noun meanings correspond to multiple topic information), which extends the scope of the verb table to cover literal topics. In addition, we find that all methods perform better on the MOX dataset than on the TroFi dataset. This may be due to the fact that the TroFi dataset contains more samples and contains a large number of pronouns and substantive nouns. In the test results on the TroFi dataset, the performance of the three topic types is relatively close, whereas on the MOX dataset, the WordNet topic and the LDA topic perform similarly, while the Oxford topic has a higher F1 score than the other two (4.8%).

Finally, it is worth noting that we observe that the performance using the topic approach is much higher than the results of metaphor detection using only GPT. This suggests that by bootstrapping and combining GPT-generated surface knowledge, such as common literal collocations of verbs, and adapting it to the domain of metaphor detection, it can significantly improve the performance of GPT in detecting verb metaphors.

In Experiment 2 (cf. Figure 1), we exclusively employ the MOH-X dataset and maintained the hyperparameter k at a fixed value of 3. The experimental findings demonstrate that augmenting the value of $k1$ results in an enhancement of the model's ability to detect metaphors, albeit to a certain extent. This improvement can be attributed to the fact that increasing $k1$ introduces a greater number of literal-meaning collocations from the verb list. Consequently, this equips the model with a better capacity to identify non-metaphorical content associated with specific verbs, thereby reducing instances of misjudgment. In addition, the performance peaks when the hyperparameter $k2$ is set to 3. However, when continuing to increase the value of $k2$, the model's performance in detecting metaphors decreases instead. This suggests that considering multiple meanings of the focal word may introduce metaphorical information or redundant topics, which may affect performance. Thus, our experimental results emphasize the need to weigh the model performance and the impact of topic introduction when choosing the value of $k2$.

5 Conclusion

We present a novel approach that aims to introduce the model knowledge of ChatGPT into the metaphor detection task. This approach does not rely on manually encoded knowledge, nor does it need to rely on manually labeled datasets. First, we construct a literal meaning collocation lookup table for each target verb. When parsing the input text, we pay special attention to the subjects and objects corresponding to the verbs to be detected. We introduce a variety of topics, including WordNet topics, LDA topics, and Oxford topics. We determine whether a text contains metaphorical expressions by comparing the relationships between subject and object topic categories in the input text and the target verb topic categories given in the verb list. The results show that by delicately combining and bootstrapping model knowledge, we are able to significantly improve the performance level of ChatGPT in the metaphor detection task.

6 Limitations

We introduce a verb table containing literal subject-verb and verb-object collocations for each target vocabulary. However, the literal collocations generated using ChatGPT are not always comprehensive, which leads to some literal samples being incorrectly categorized as metaphorical usage. In addition, due to varying syntactic structures, when analyzing subject-verb-object relations in input texts using ChatGPT, there may be parsing errors or structures that are not present, which also affects the performance of the overall method. In future work, we would like to investigate more powerful generative models or natural language parsing tools to improve the coverage of literal collocations in verb lists or to improve the accuracy of parsing subject-verb-object relations of input texts.

7 Ethics Statement

Metaphor, as a linguistic phenomenon that conveys implicit semantics, is capable of concretizing abstract concepts or enriching substantive concepts.

This makes it possible for metaphors to be used as a tool for communicating political positions and gaining voter support in the political domain. However, our proposed zero-shot metaphor detection approach can also be used to identify metaphorical expressions and address the above issues from a governance perspective. In addition, we advocate the inclusion of tasks related to metaphor detection and generation, especially the application of ChatGPT to downstream metaphor applications, into the AI ethical code.

References

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.

Julia Birke and Anoop Sarkar. 2006. A clustering approach for nearly unsupervised recognition of nonliteral language. In *11th Conference of the European Chapter of the Association for Computational Linguistics*, pages 329–336.

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

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. *arXiv preprint arXiv:2104.13615*.

Lisa Gandy, Nadji Allan, Mark Atallah, Ophir Frieder, Newton Howard, Sergey Kanareykin, Moshe Koppel, Mark Last, Yair Neuman, and Shlomo Argamon. 2013. Automatic identification of conceptual metaphors with limited knowledge. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 27, pages 328–334.

David Graff, Junbo Kong, Ke Chen, and Kazuaki Maeda. 2003. English gigaword. *Linguistic Data Consortium, Philadelphia*, 4(1):34.

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

Ilana Heintz, Ryan Gabbard, Mahesh Srivastava, Dave Barner, Donald Black, Majorie Friedman, and Ralph Weischedel. 2013. Automatic extraction of linguistic metaphors with lda topic modeling. In *Proceedings of the First Workshop on Metaphor in NLP*, pages 58–66.

Mary Hesse. 1965. Models and analogies in science.

Yael Karov and Shimon Edelman. 1998. Similarity-based word sense disambiguation. *Computational linguistics*, 24(1):41–59.

Adam Kilgarriff. 2000. Wordnet: An electronic lexical database.

Dan Klein and Christopher D Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of the 41st annual meeting of the association for computational linguistics*, pages 423–430.

Saisuresh Krishnakumaran and Xiaojin Zhu. 2007. Hunting elusive metaphors using lexical resources. In *Proceedings of the Workshop on Computational approaches to Figurative Language*, pages 13–20.

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

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*. Association for Computational Linguistics (ACL).

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

Malay Pramanick and Pabitra Mitra. 2018. Unsupervised detection of metaphorical adjective-noun pairs. In *Proceedings of the Workshop on Figurative Language Processing*, pages 76–80.

KC Shet, U Dinesh Acharya, et al. 2012. A new similarity measure for taxonomy based on edge counting. *arXiv preprint arXiv:1211.4709*.

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*, pages 160–170.

Ekaterina Shutova and Lin Sun. 2013. Unsupervised metaphor identification using hierarchical graph factorization clustering. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 978–988.

Ekaterina Shutova, Simone Teufel, and Anna Korhonen. 2013. Statistical metaphor processing. *Computational Linguistics*, 39(2):301–353.

Ekaterina Shutova, Tim Van de Cruys, and Anna Korhonen. 2012. Unsupervised metaphor paraphrasing using a vector space model. In *Proceedings of COLING 2012: Posters*, pages 1121–1130.

Chuandong Su, Fumiyo Fukumoto, Xiaoxi Huang, Jiyi Li, Rongbo Wang, and Zhiqun Chen. 2020. Deepmet: A reading comprehension paradigm for token-level metaphor detection. In *Proceedings of the second workshop on figurative language processing*, pages 30–39.

Yulia Tsvetkov, Elena Mukomel, and Anatole Gershman. 2013. Cross-lingual metaphor detection using common semantic features. In *Proceedings of the First Workshop on Metaphor in NLP*, pages 45–51.

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

Lennart Wachowiak and Dagmar Gromann. 2023. Does gpt-3 grasp metaphors? identifying metaphor mappings with generative language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1018–1032.

Shun Wang, Yucheng Li, Chenghua Lin, Loïc Barrault, and Frank Guerin. 2023. Metaphor detection with effective context denoising. *arXiv preprint arXiv:2302.05611*.

Yorick Wilks, Adam Dalton, James Allen, and Lucian Galescu. 2013. Automatic metaphor detection using large-scale lexical resources and conventional metaphor extraction. In *Proceedings of the First Workshop on Metaphor in NLP*, pages 36–44.

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