MetaPro 2.0: Computational Metaphor Processing on the Effectiveness of Anomalous Language Modeling

Rui Mao[♠], **Kai He**[♣], **Claudia Beth Ong**[♠], **Qian Liu**[♦] and **Erik Cambria**[♠]
[♠]Nanyang Technological University, Singapore
[♣]National University of Singapore, Singapore
[♦]University of Auckland, New Zealand
{rui.mao, cambria}@ntu.edu.sg, kai_he@nus.edu.sg,
clau0011@e.ntu.edu.sg, liu.qian@auckland.ac.nz

Abstract

Metaphor interpretation is a difficult task in natural language understanding. The development of relevant techniques in this domain is slow, mostly because of the lack of large annotated datasets and effective pre-trained language models (PLMs) for metaphor learning. Thus, we propose a large annotated dataset and a PLM for the metaphor interpretation task. Our foundation model is based on a novel anomalous language modeling (ALM) method, which we benchmark with comparable PLM baselines on the new dataset, finding that it largely improves model performance on metaphor identification and interpretation.

1 Introduction

A metaphor is defined as using one (a Single-Word Expression, SWE) or several words (a Multi-Word Expression, MWE) to represent a different meaning, rather than its literal meaning (Lagerwerf and Meijers, 2008). Interpreting the meanings of metaphors in contexts is particularly challenging for machines (Stowe et al., 2022). There are two important tasks in linguistic metaphor processing, i.e., metaphor identification and interpretation. The former task is normally defined as a classification task, detecting metaphors on a sentence level (Heintz et al., 2013), a word-pair level (Ge et al., 2022), or a token level (Stowe et al., 2019). The latter task is normally defined as a property extraction task (Su et al., 2020), a paraphrasing task (Mao et al., 2018), or an explanation pairing task (Mao et al., 2022). Token-level metaphor identification, and paraphrasing-based metaphor interpretation are more supportive for natural language processing (NLP) downstream tasks, because metaphors can be paraphrased into their literal counterparts without breaking the coherence and general syntax of the original sentences, e.g., "I don't *buy*[1] your story" vs "I don't believe your story".

[1]Italics denote metaphors.

Machines can directly use paraphrased sequences to improve metaphor understanding in downstream tasks. Steen et al. (2010) have developed the largest token-level metaphor identification dataset, VU Amsterdam Metaphor Corpus (VMC). Many previous metaphor identification models were developed upon it (Choi et al., 2021; Mao and Li, 2021; Li et al., 2023a,b), advancing the task significantly. In contrast, the progress of metaphor interpretation has been falling behind. This is likely because of the lack of a large paraphrasing-based metaphor interpretation dataset and an effective pre-trained language model (PLM) for the learning of metaphor interpretation. We are motivated to develop a metaphor interpretation-orientated dataset, VU Amsterdam Metaphor Corpus with Paraphrases (VMC-P)[2], and a new PLM with a novel anomalous language modeling (ALM)[3] pre-training paradigm.

We aim to develop a dataset for training end-to-end (E2E) metaphor interpretation systems. We believe that a good metaphor paraphrase dataset should contain sufficient reliable training data; Metaphors in the dataset should be used in a manner similar to everyday language. Compared to the latest metaphor interpretation dataset (IMPLI) (Stowe et al., 2022) (to the best of our knowledge, it may be also the largest) which contains 920 manually paraphrased metaphors and idioms and 17,027 automatically paired paraphrases, our VMC-P dataset has more manually annotated metaphor paraphrases, accounting for 11,880 units, covering both SWEs and MWEs. Besides, each sentence in IMPLI only has a paraphrase for a single target metaphor, which is dissimilar to the real-world distribution. In contrast, the sentences in our dataset can have multiple metaphors and corresponding paraphrases (2.2 metaphor units per metaphorical sentence on average).

[2]https://huggingface.co/datasets/RuiMao1988/VMC-P
[3]https://huggingface.co/RuiMao1988/ALM

Findings of the Association for Computational Linguistics ACL 2024, pages 9891–9908

In real-world texts, it is common for a sentence to have multiple metaphors (see the example in Figure 4). Thus, our dataset is closer to real-world scenarios. There is another metaphor paraphrase dataset (Bizzoni and Lappin, 2018), while it just contains 200 gold paraphrases. Our proposed pre-training paradigm, ALM, is metaphor processing-tailored and linguistics-informed, compared to current masked word prediction (MWP)-based PLMs, e.g., BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019), language modeling-based PLMs, e.g., GPTs (Radford et al., 2018, 2019; Brown et al., 2020), and sequence-to-sequence-based PLMs, e.g., T5 (Raffel et al., 2020). ALM is motivated by the fact that metaphors do not take their literal meanings in contexts. We hypothesize that the literal meanings of metaphoric words are likely contextually anomalous. For example, given "my car *drinks* gasoline", literally, neither a car can drink, nor can gasoline be drinkable. Wilks (1975, 1978) explained this as Selectional Preference Violation (SPV) – metaphors violate the selectional preference of their contexts. We hypothesize that a randomly replaced word also likely violates the selectional preference of its context. Thus, we can develop a pre-training corpus with randomly replaced words to simulate the SPV of metaphors.

ALM aims to detect anomalous replacements and retrieve original words to simulate metaphor identification and interpretation tasks, respectively. The pre-training corpus is free of human annotation as we can automatically replace an SWE/MWE with a random one, achieving a large pre-training corpus with many anomalies. We use contrastive learning to simulate a human Metaphor Identification Procedure (MIP) (see Appendix B, Pragglejaz, 2007): A metaphor is identified based on the semantic contrast between its contextual and basic meanings. Contrastive learning of positive and hard negative samples also helps ALM to learn the boundary between them, which is an important human knowledge acquisition process (NASEM, 2018).

We re-train RoBERTa-large with pre-processed WIKIPEDIA that contains hard anomalous SWEs and MWEs. We compare the re-trained model, also termed ALM, with parameter size-comparable PLMs, e.g., large BERT, RoBERTa, medium GPT2, and T5-base, finding that ALM exceeds the best PLM by 3.20% and 3.60% averaged F1 in metaphor identification and E2E interpretation tasks, based on VMC-P dataset. The fine-tuned ALM for metaphor processing is termed MetaPro 2.0.

We hope the preliminary success of our pre-training paradigm in a small PLM may inspire future large language model (LLM) development, although our model may not be able to exceed them in the same fine-tuning setups. Our contribution is twofold: (1) We propose a new dataset for training E2E metaphor interpretation systems. To our knowledge, this is the largest manually developed, token-level metaphor paraphrase dataset. (2) We propose a PLM with a novel ALM paradigm for metaphor interpretation, which outperforms previous PLMs with similar parameter sizes.

2 Related Work

Metaphor identification is a well-studied area, because of rich annotated data resources (Birke and Sarkar, 2006; Steen et al., 2010; Mohammad et al., 2016; Xu et al., 2022). Currently, the token-level metaphor identification task is likely formulated as a sequence tagging task (Stowe et al., 2019; Chen et al., 2021; Li et al., 2021), using PLMs, e.g., BERT and RoBERTa. Thus, an effective PLM is important for the success of metaphor detection.

Metaphor interpretation is underdeveloped. Su et al. (2020) interpreted metaphors by extracting properties that were shared by source and target domains. Bizzoni and Lappin (2018) paired metaphoric sentences with handwritten paraphrases on the sentence-level. Mao et al. (2018) paraphrased metaphors on the token level without testing MWEs. Mao et al. (2022) proposed an E2E model for identifying and interpreting metaphoric SWEs and MWEs. The SWE interpretation was given by the original RoBERTa-based MWP without fine-tuning, due to the lack of labeled data. They introduced a metaphoric MWE dictionary to explain a metaphor via a clause after an original sentence. Mao et al. (2023b) integrated a concept mapping method, based on metaphor and its paraphrase (Mao et al., 2022), showing the utility of such a metaphor processing paradigm in cognitive analysis (Han et al., 2022; Mao et al., 2023a, 2024b). However, Mao et al. (2022) have some limitations including gaps between the probability distributions of the raw RoBERTa MWP and the distributions of metaphor paraphrases in contexts.

Metaphor interpretation datasets were built for tasks such as definition-pairing (Zayed et al., 2020), sentence-level paraphrases (Bizzoni and Lappin, 2018), and natural language inference (NLI) testing (constitution-level) (Stowe et al.,

2022; Chakrabarty et al., 2022). The definition-pairing-based dataset includes metaphor definitions from dictionaries, which are independent of contexts. Unlike sentence-level interpretation datasets that paraphrase the meaning of a metaphoric sentence as a whole, finer-grained interpretation datasets (Stowe et al., 2022; Chakrabarty et al., 2022) paraphrase metaphoric SWEs and MWEs within contexts, and does not break up the general syntactical structures of original sentences, because the literal context words are not changed after paraphrasing. Thus, finer-grained interpretation likely yields more context-consistent paraphrases and reduces the uncertainty of the learning for machines. Since Stowe et al. (2022); Chakrabarty et al. (2022) aimed to study figurative NLI, each target sentence contains just an in-context SWE/MWE paraphrase, which does not link well with everyday language where a sentence can have multiple metaphors. Besides, these datasets include clear-cut figurative examples for NLI testing, thus, they may be sub-optimal for training a metaphor processing model for paraphrasing real-world texts (Shutova, 2015).

PLMs are critical for metaphor identification and interpretation. Researchers developed different pre-training tasks and architectures to achieve contextualized representations for input sequences, e.g., MWP tasks (BERT and RoBERTa), language modeling tasks (GPT family), and sequence-to-sequence learning tasks (T5). These PLMs were trained for general learning purposes, whereas the pre-training task setups were not tailored by the linguistic intuition of metaphors. For example, language modeling and sequence-to-sequence learning methods did not take bi-directional contexts into account. MWP methods used a special token "[MASK]" to replace one or several original tokens to learn their representations from bi-directional contexts, whereas, MWP ignored the semantics of the original masked-out words, simply learning from their contexts. These setups are sub-optimal for the learning of metaphor processing because metaphors provide necessary semantic information in the context for their interpretation; both the forward and backward contexts help to identify metaphors and interpret their intended meanings.

3 VMC-P Dataset Development

We aim to develop a metaphor interpretation dataset that helps to yield E2E systems for real-world application scenarios. We define the interpretation task as metaphoric SWE and MWE paraphrasing (token-level interpretation). Our dataset contains 10,716 sequences, where 50.12% of them are non-literal. We have paraphrased 11,880 metaphor units (SWEs and MWEs). The detailed statistics and an example of our dataset can be viewed in Appendix C.

We source data from VMC, as it is the largest all-word annotated metaphor identification dataset. The VMC data were sourced from the everyday language with academic, fiction, news, and conversation genres. We focus on labeling open-class metaphors as they deliver richer semantic information. Their paraphrases will benefit more downstream tasks. If a metaphoric MWE unit includes words with other Parts-of-Speech (PoS), we paraphrase the MWE as a whole. Metaphor identification serves interpretation in our dataset. If a metaphor cannot be paraphrased within its context without breaking the precision and conciseness of language, it is not labeled as a metaphor. We cancel 8,503 (40.2%) original metaphor labels from VMC, including 6,895 (32.6%) closed-class metaphors.

Metaphor interpretation is subjective and creative (Indurkhya, 2007), which is the key annotation challenge. To improve labeling consistency, we took the following measures: First, we employed an English-speaking expert with a linguistic education background to annotate the full dataset[4]. Thus, the only annotator can generate consistent paraphrases for metaphors[5]. Second, we used WordNet (Fellbaum, 1998) hypernyms and synonyms as annotation references[6]. We developed a dictionary ($\mathcal{D}$) that allows a user to query hypernyms and synonyms of semantic units. The annotator was asked to choose the most appropriate paraphrase lemma with the same PoS as the original metaphor from $\mathcal{D}$. If a suitable option was not available, the annotator then wrote a paraphrase by his/her own understanding. The references from the dictionary help the annotator generate consistent paraphrases. Besides, using the dictionary as a knowledge base is computation-friendly.

[4]The Singaporean annotator and evaluators noticed the task is a part-time job for NLP research, paying $9 per hour (local part-time rate is $9: https://www.mom.gov.sg/employment-practices/progressive-wage-model/local-qualifying-salary). They were asked to take a break every 30 minutes.

[5]Introducing multi-annotators may lead to increased variability in our annotation task, making it more difficult to establish definitive ground truth. Thus, we use an annotate-then-evaluate process. The rationality is discussed in Appendix A.

[6]When it comes to the annotation and machine learning of non-English metaphors, one can use BabelNet (Navigli and Ponzetto, 2012) as a structured resource of synonyms.

3.1 Metaphor interpretation annotation

The annotator was trained with MIP (see Appendix B) first. After achieving at least 90% correct metaphoric paraphrases (including correctly identifying metaphors) out of 50 instances, the annotator started to annotate the paraphrases in VMC-P.

First, the annotator would read a text from VMC, and understand the overall meaning. Then, the annotator would interpret the contextual meaning and basic meaning of a given metaphor in the text. If the contextual meaning contrasts with the basic meaning, and the contextual meaning can be understood in a comparable way, the annotator would generate a paraphrase whose basic meaning can represent the contextual meaning of the metaphor as the gold paraphrase. The generated paraphrase should be coherent with the context and use the same word form as the original metaphor. If the contextual and basic meanings are not contrastive, or there is no better paraphrase to represent the metaphor, the annotator would mute the original metaphoricity label. Such a case likely appears in dead metaphors, e.g., "*falling in* love". Cognitively, the metaphoricity of "*falling in*" is given by the fact that the source and target concepts are from different domains, e.g., LOVE IS SPACE. However, it is hard to find a paraphrase whose basic meaning represents the contextual meaning of "*falling in*" and keep the language concise and precise. Paraphrasing these metaphors is sub-optimal for using metaphor processing for downstream task pre-processing, so we did not paraphrase them. We also canceled the metaphoricity of scientific words, e.g., "*sausage* instability" and "electric *field*", because these words have been widely recognized and accepted by the public. We cannot find better paraphrases for them.

The annotator was advised to generate a single-word replacement as a paraphrase for easier computing. If a single-word replacement loses the emotional connotation of the original metaphor, an additional modifier is added, e.g., paraphrasing a verbal metaphor with an adverb-verb phrase ("Adam *snarled*" → "Adam angrily said"). If a metaphor is an MWE, e.g., idioms and prepositional phrases, the annotator would consider the consecutive words in the text as a whole for paraphrasing ("*shaken a large fist*" → "demonstrated the anger"; "*filtering out*" → "separating"). A simile would be paraphrased as another simile, while the paraphrase would be closer to the intended meaning of the original simile ("like a *snake*" → "like a bad person"). The annotator was advised to sparingly generate WordNet uncovered paraphrases to maintain annotation consistency. Any newly generated paraphrases were automatically added to the reference dictionary $\mathcal{D}$ for later references.

After obtaining the gold paraphrase (positive sample) of a metaphor, the corresponding negative samples are also included in our dataset, which are the WordNet hypernyms and synonyms of the original metaphor that were not selected by the annotator. They are sub-optimal paraphrases for the annotator. Negative samples of WordNet uncovered SWEs or MWEs were sourced from Google by searching their synonyms. Negative samples also include annotator-generated paraphrases for other metaphors with the same lemma as the current metaphor, while the paraphrases differ from the current gold paraphrase. Thus, one can use those samples in contrastive learning and evaluate a metaphor interpretation model by measuring if the model can select the gold paraphrase from the collection of positive and negative samples. Given a PoS, if the number of negative samples is lower than three, negative samples will additionally include the hypernyms and synonyms of the metaphor from other PoS, so that we can collect sufficient hard negative samples for contrastive learning. The negative samples are hard samples as they are semantically similar to the original metaphor in different senses, and most of them have the same PoS and word forms as the original metaphor.

3.2 Quality control

We invited two native English speakers from Singapore to evaluate all the expert-annotated 11,880 gold paraphrases. They evaluated if a given paraphrase of a metaphor is acceptable in the context by the annotation criteria. The average accuracy voted by the two raters is 99.79%. 99.64% paraphrases were rated as "acceptable" by both raters (agreed by the three participants); 0.30% were "acceptable" for one of the raters (agreed by the two participants); 0.07% were "unacceptable" for both raters. We re-evaluated the paraphrases disagreed by both raters and corrected the corresponding errors.

4 ALM Pre-training and Fine-tuning

A new PLM is also proposed in this work, based on a novel ALM paradigm. ALM-based pre-training is inspired by two linguistic findings about metaphors: (1) the literal meaning of a metaphor violates the

selectional preference of its context (SPV); (2) a metaphor can be identified if there is a semantic contrast between its contextual and basic meanings (MIP). Finding 1 motivates us to develop a pre-training corpus with anomalous word replacements that violate the selectional preference of contexts. Then, a PLM learns to detect anomalies. Finding 2 motivates us to use contrastive learning to learn the semantic contrast between the real meaning of a word and its other meanings. Then, the PLM learns to retrieve the original words by pulling the representation of a replaced word close to the representation of the original word (real meaning) and pushing the representation of the replaced word distant to the representations of its other synonyms (other meanings). We use the synonym of an original word instead of [MASK] to represent the anomaly, because we assume that the synonym replacement has necessary semantic information in the context, which is similar to a metaphor. The above setups simulate metaphor identification and interpretation tasks, respectively. Thus, a PLM can learn more useful task-specific knowledge for computational metaphor processing via the ALM paradigm, as opposed to the MWP, language modeling, and sequence-to-sequence learning paradigms.

To support both metaphor identification and interpretation tasks, our pre-training employs a multi-task learning (MTL) framework to learn the anomaly detection task (a sequence tagging task corresponding to metaphor identification in fine-tuning) and contrastive learning task (a word classification task corresponding to metaphor interpretation in fine-tuning) together.

4.1 Pre-training corpus preparation

We use English WIKIPEDIA as the pre-training data source. We aim to develop a pre-training corpus that simulates the SPV of metaphors and supports the learning of semantic contrasts between the real meaning of a word and its other meanings within a context. WIKIPEDIA is used because it likely uses literal expressions to introduce scientific knowledge with formal language. It was also used by other PLMs (Devlin et al., 2019; Liu et al., 2019). Thus, aside from implementing a novel pre-training paradigm, we did not incorporate additional corpora to enhance the pre-training outcomes.

We hypothesize that the meaning of a word can be symbolically represented by one of its synonyms and hypernyms under a specific sense. For example, the verb "buy" has several senses in WordNet, e.g., "obtaining by purchase", "accepting as true", etc. Given the first sense, "buy" equals "purchase", e.g., "I will buy/purchase a new laptop". Given the second sense, "buy" equals "believe", e.g., "I can't *buy*/believe this story". Thus, the sense of "buy" should be closer to the sense of "believe" than "purchase" in the context of "I can't *buy* your story". Given "I can't believe your story", "believe" satisfies the selectional preference in the context. If we replace "believe" with another word, the word likely violates the selectional preference.

We randomly select a synonym of an original word from WordNet to replace the original word, where the replacement is defined as an anomalous word in a context[7]. Synonyms are mutual, e.g., "buy" is the synonym of "believe", and vice versa. Then, in the later fine-tuning stage, the real meaning of a metaphor can also be retrieved from one of its candidates[8]. The original word is considered as the positive meaning of the anomalous word. Next, we sample n (a hyperparameter) synonyms of the anomalous word, excluding the original word, as negative meanings.

Our pre-training task is to detect the anomalous word and retrieve the original word (the real meaning; a positive sample) out of other synonyms of the anomalous word (other meanings; negative samples). We randomly select 35% open-class words of a sentence to replace them to develop a pre-training sequence. The criterion of selecting the original word for replacing is that the frequency of the lemma of a selected word is above the median within WIKIPEDIA so that we can collect sufficient positive samples with different contexts for contrastive learning. The anomalous words, positive and negative words can also be MWEs because WordNet contains MWEs. If there is no word satisfying the criterion, the sentence is not included in the pre-training corpus. To obtain hard negative samples and anomalous words, their word forms are aligned to the original word. The word form

[7]For non-English pre-training, one can also use word embeddings for semantically similar replacement acquisition.

[8]We include synonyms and hypernyms as candidates for VMC-P dataset development and fine-tuning, because hypernyms also have similar meanings as their covered words, although the hypernym meanings are relatively more abstract than synonyms. More candidates (using both synonyms and hypernyms) yield more annotation references and negative samples for fine-tuning. We do not include hypernyms as negative samples for pre-training, because we can achieve sufficient pre-training data with synonyms only. Furthermore, synonyms may be harder negative samples than hypernyms for more effective contrastive learning-based pre-training.

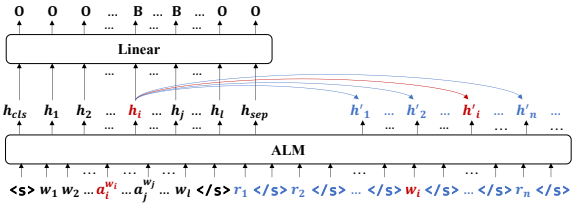


Figure 1: The MTL framework of our model for ALM-based pre-training and fine-tuning. O denotes a negative label, e.g., the label of a non-anomalous word or a literal word; B denotes a positive label, e.g., the label of an anomalous word or a metaphoric word. h is a hidden state of ALM outputs. a denotes an anomalous word or a metaphor. The red w_i in the prompt denotes an original word or a gold paraphrase, corresponding to $a_i^{w_i}$. r denotes a negative sample, corresponding to $a_i^{w_i}$.

alignment method is from Mao et al. (2022). To obtain the frequency statistics and the word forms of anomalous words and negative samples, we run PoS tagging and lemmatization on full WIKIPEDIA.

4.2 ALM-based pre-training

Our pre-training is upon RoBERTa-large[9] and Byte-Pair Encoding (BPE) tokenizer (Radford et al., 2019), due to its effectiveness in metaphor detection (Leong et al., 2020). Given a text sequence with m anomalous words (a), l original words (w), and n randomly sampled negative words (r) of $a_i^{w_i}$, a pre-training input (s) is organized as $s =$ <s>$w_1 w_2 ... a_i^{w_i} ... a_j^{w_j} ... w_l$</s>$r_1$</s>$r_2$</s>...</s> w_i</s>...</s>r_n</s>, where $a_i^{w_i}$ denotes the word or MWE at Position i is anomalous, corresponding to the original word or MWE w_i. <s> and </s> are special tokens. An input sequence can have multiple anomalous words and a prompt. We define the sequence after the first </s> as a prompt (the blue input tokens in Figure 1), including a positive sample (w_i) and n negative samples of $a_i^{w_i}$. The position of w_i is random in the prompt.

An MTL framework is used for pre-training (Figure 1). For each training step, ALM learns to detect all anomalous words via a sequence tagging task and retrieve the original word (w_i) of an anomalous word ($a_i^{w_i}$) out of $n+1$ candidates via a prompt and contrastive learning. Learning the next original word (w_j) of the sequence is in a different training step with another prompt. $a_i^{w_i}$ is an anchor, pulling its representation close to the representation of w_i and pushing its representation distant to the representations of negative samples via contrastive learning. Thus, the model can learn the semantic contrast between the real meaning and other meanings within a context. If $a_i^{w_i}$, w_i or r has multi-tokens, we average the hidden states (h in Figure 1) of all the tokens for contrastive learning. Upon ALM, we use a linear layer to identify all anomalous words. We use a cross-entropy loss to learn the sequence tagging task ($\mathcal{L}^{(seq)}$)

$$\mathcal{L}^{(seq)} = \text{CrossEntropy}(\hat{Y}^{(seq)}, Y^{(seq)}).$$

The contrastive learning loss ($\mathcal{L}^{(con)}$) is from He et al. (2020) with an Euclidean distance measure

$$\mathcal{L}^{(con)} = -\sum_i \log \frac{exp(E(h_i, h'_i)/\tau)}{\sum_{j\in\{1,...,n\}} exp(E(h_i, h'_j)/\tau)},$$

where $E(\cdot)$ denotes Euclidean distance. h and h' denote the hidden states of anomalous sentences and prompts from the ALM output, respectively. τ denotes a temperature hyperparameter. The overall loss ($\mathcal{L}$) is the weighted sum of the learning tasks

$$\mathcal{L} = \alpha\mathcal{L}^{(seq)} + \beta\mathcal{L}^{(con)},$$

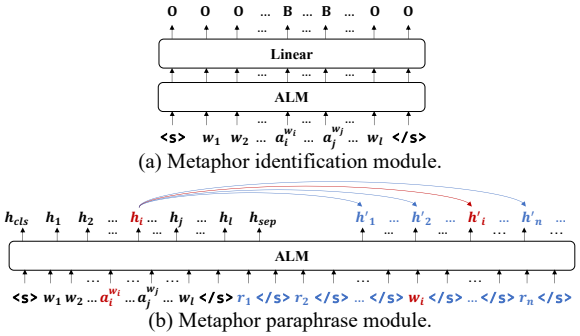
where α and β are hyperparameters for pre-training ALM on WIKIPEDIA.

Compared to MWP-based pre-training, the advantages of ALM are summarized as follows: (1) ALM incorporates the semantics of an anomalous word (a synonym of the original word) to infer the original word, while MWP uses a unified [MASK] to predict the original word, although both methods use bi-directional contexts. (2) The contrastive learning allows ALM to take the semantics of the inferred positive and negative samples into account via an additional input prompt, while MWP considers the original word (a positive sample) as an output label. Taking the semantics of the predicted label words into account is helpful for learning MWEs because different word combinations may have similar or different meanings, e.g., "take", "take in", "take in charge" and "take in water". (3) The predicted probability distributions of contrastive learning narrow down to the scope of limited candidates in a prompt, while MWP probability distributions cover the whole vocabulary. The complexity of ALM reasoning is greatly reduced.

4.3 Fine-tuning and testing

During the fine-tuning stage, ALM learns to identify metaphors (a sequence tagging task) and retrieve the gold paraphrase of a metaphor out of

[9] https://huggingface.co/FacebookAI/roberta-large



(a) Metaphor identification module.

(b) Metaphor paraphrase module.

Figure 2: The STL framework of MetaPro 2.0.

$k+1$ candidates (a contrastive learning-based word classification task). Then, in Figure 1, $a_i^{w_i}$ is a metaphor whose gold paraphrase is w_i. r is a negative sample. We select k negative samples for each fine-tuning step from the reference dictionary $\mathcal{D}$. We examine single-task learning (STL) and MTL frameworks, respectively, because STL is the most popular framework for metaphor identification (Ge et al., 2023); MTL is our pre-training framework.

For STL (see Figure 2), we use two ALMs to learn metaphor identification and paraphrase tasks, separately, with separated loss functions (a cross-entropy loss $\mathcal{L}^{(seq)}$ for sequential metaphor identification, and a contrastive learning loss $\mathcal{L}^{(con)}$ for metaphor paraphrase detection). The metaphor identification module learns a sequence without a prompt, while the metaphor paraphrase module learns a sequence with a prompt and a target metaphor. For MTL, we use ALM to learn both tasks together, which is similar to the pre-training framework and input sequence structure in Figure 1. The MTL overall loss for fine-tuning is defined as

$$\mathcal{L}' = \alpha'\mathcal{L}^{(seq)} + \beta'\mathcal{L}^{(con)}.$$

During the MTL testing stage, we first feed a text with a prompt containing $k+1$ <pad> tokens for querying a metaphor identification output. The <pad> tokens are separated by </s>. Next, we lemmatize an identified metaphor and sample its $k+1$ candidate paraphrases from the reference dictionary $\mathcal{D}$. The word forms of the selected candidates are aligned with the target metaphor by universal dependencies (Nivre et al., 2016), then fed into the model via a prompt. Since the $k+1$ sampled words may be less than the total number of candidates of the target metaphor in $\mathcal{D}$, we sample multiple times and compare them until finding the winner that is the most similar to the metaphor in vector space among all the candidates. The winner is the predicted paraphrase of the target metaphor. If multiple metaphors are identified, the model predicts multiple times until each metaphor has a paraphrase. Testing the STL models is similar. The only difference is we directly feed testing data into the metaphor identification model without concatenating a prompt to query an identification output.

5 Experiments

Baselines. ALM has 355M parameters. We benchmark ALM to other parameter size-comparable PLMs, e.g., BERT-large-cased (340M), RoBERTa-large (355M), GPT2-medium (345M), and T5-base (220M), based on the MTL and STL frameworks in Figures 1 and 2. Baseline hyperparameters are tuned by the results of the validation set.

State-of-the-art metaphor processing models are not our direct baselines, because ***a***) their technical novelties are irrelevant to pre-training paradigms; ***b***) they are not foundation models for computational metaphor processing; ***c***) our work focuses on E2E metaphor interpretation rather than metaphor identification; ***d***) the only E2E metaphor paraphrase baseline (Mao et al., 2022) cannot paraphrase metaphoric MWEs in context and was not fine-tuned with metaphor interpretation datasets[10], whereas these are the advancement of the proposed ALM and VMC-P. LLMs are also not comparable to ALM because of their huge amount of parameters, significant training, and inference costs. However, due to the absence of evaluating LLMs on E2E metaphor interpretation, we test GPT-3.5/4 on VMC-P in Appendix E.

Pre-training Setups. ALM is pre-trained on 4 NVIDIA RTX A6000 GPUs for 21 days. The batch size is 512. The number (n) of negative samples is 5. We use Adam optimizer (Kingma and Ba, 2015) with a learning rate of 1e-5. The contrastive learning loss temperature (τ) is 0.05. The pre-training loss weights (α and β) are 10 and 1, respectively.

Fine-tuning Setups. ALM is fine-tuned on an NVIDIA V100 GPU. The batch size is 4. The number (k) of negative samples is 3. We use Adam optimizer with a learning rate of 1e-5. The contrastive learning loss temperature (τ) is 0.05. The fine-tuning loss weights (α' and β') are 10 and 1, respectively. Fine-tuning early stops by 5 epochs, examined on the validation set. The reported results are averaged over 5 runs.

[10]The fine-tuned metaphor paraphrase module of Mao et al. (2022) equals to the RoBERTa-large baseline.

Setup	Task	Model	P	R	F1
STL	Identification	BERT	66.98	64.85	65.88
		RoBERTa	69.87	65.98	67.83
		GPT2	52.17	63.21	57.14
		T5	62.70	61.25	61.94
		MetaPro2.0	**72.41**	**68.25**	**70.23***
	Interpretation (gold idnt. label-based)	BERT-G	26.70	100	42.13
		RoBERTa-G	28.02	100	43.76
		GPT2-G	24.51	100	39.35
		T5-G	25.87	100	41.09
		MetaPro2.0-G	**30.28**	100	**46.46***
	Interpretation (predicted idnt. label -based)	BERT-P	23.26	66.91	34.51
		RoBERTa-P	24.45	65.72	35.62
		GPT2-P	19.18	63.84	29.46
		T5-P	21.94	58.73	31.92
		MetaPro2.0-P	**27.63**	**69.73**	**39.55***
MTL	Identification	BERT	65.72	65.40	65.53
		RoBERTa	67.25	65.23	66.20
		GPT2	57.45	57.94	57.68
		T5	63.49	63.02	63.24
		MetaPro2.0	**70.28**	**70.17**	**70.20***
	Interpretation (gold idnt. label-based)	BERT-G	26.11	100	41.39
		RoBERTa-G	26.94	100	42.43
		GPT2-G	23.56	100	38.11
		T5-G	24.49	100	39.32
		MetaPro2.0-G	**29.02**	100	**44.97***
	Interpretation (predicted idnt. label -based)	BERT-P	22.57	63.69	33.30
		RoBERTa-P	24.18	63.72	35.02
		GPT2-P	19.47	50.27	28.05
		T5-P	21.41	57.76	31.21
		MetaPro2.0-P	**26.49**	**69.11**	**38.28***

Table 1: Overall performance, evaluated on a unit level. Precision (P), Recall (R) and F1 scores are averaged over 5 runs separately. * denotes the improvement is statistically significant ($p < 0.001$) on a two-tailed test.

PoS	Model	STL		MTL	
		$\mathbf{Acc}_{\mathrm{id}}$	$\mathbf{Acc}_{\mathrm{in}}$	$\mathbf{Acc}_{\mathrm{id}}$	$\mathbf{Acc}_{\mathrm{in}}$
VB	BERT	86.47	23.29	86.69	24.76
	RoBERTa	87.94	26.56	87.12	27.28
	GPT2	81.91	23.10	79.42	20.41
	T5	85.05	25.66	84.85	22.82
	MetaPro2.0	**89.58**	**29.35**	**89.33**	**28.58**
NN	BERT	91.23	29.99	91.24	28.19
	RoBERTa	92.14	30.78	91.93	31.30
	GPT2	89.76	26.14	88.47	26.72
	T5	91.18	28.42	91.36	26.36
	MetaPro2.0	**92.73**	**32.00**	**93.21**	**31.72**
ADJ	BERT	84.33	32.07	86.11	28.90
	RoBERTa	86.19	35.50	85.49	35.01
	GPT2	82.27	26.41	79.94	26.78
	T5	84.81	30.74	85.28	30.30
	MetaPro2.0	**87.89**	**39.68**	**87.45**	**37.71**
ADV	BERT	94.69	28.30	94.80	29.56
	RoBERTa	95.00	35.22	94.15	33.10
	GPT2	93.71	26.75	92.79	23.57
	T5	93.77	24.19	94.24	30.65
	MetaPro2.0	**95.55**	**35.28**	**95.39**	**33.21**

Table 2: Breakdown analysis on a token level and the testing set. Metaphor interpretation (in.) is based on gold metaphor identification (id.) labels.

Label schema. We use BIO schema for metaphor identification labels. B denotes the beginning token of a BPE tokenized metaphor (SWE or MWE); I denotes inside the metaphor; O denotes a literal. We use F1 as the main measure for identification and interpretation tasks (see Appendix D for evaluation metrics and computational details).

6 Results

We evaluate fine-tuned ALM, termed MetaPro 2.0, with STL and MTL, based on VMC-P. More sequence tagging evaluation tasks can be viewed in Appendix F. We show the metaphor interpretation results with gold (G) and predicted (P) identification labels. Since we aim to offer useful resources for E2E metaphor interpretation, metaphor interpretation performance with predicted metaphor identification labels is the primary concern.

6.1 Overall performance

As shown in Table 1, MetaPro 2.0-P achieves large gains in F1 scores over the strongest baseline (RoBERTa-P) on the E2E metaphor interpretation task with predicted metaphor identification labels, under STL (+3.93%) and MTL (+3.26%) setups. This indicates the effectiveness of ALM pre-training paradigm, because we did not use additional pre-training corpora compared to RoBERTa. Compared to non-MWP-based PLMs, the improvements over GPT2 (STL: +10.09; MTL: +10.23%) and T5 (STL: 7.63%; MTL: 7.07%) are larger, indicating the utility of bi-directional context learning in our pre-training. The improvements under the E2E setup can also be attributed to the accuracy gains of MetaPro 2.0 in metaphor identification and gold identification label-based metaphor interpretation. For example, MetaPro 2.0 outperforms RoBERTa by 2.4% (STL) and 4.0% (MTL) in metaphor identification. MetaPro 2.0-G outperforms RoBERTa-G by 2.70% (STL) and 2.54% (MTL) in gold identification label-based metaphor interpretation. STL models often outperform MTL models due to the absence of sophisticated encoders and soft-parameter sharing mechanisms that optimize the learning of task-specific features in MTL. Finally, there is a huge performance gap between metaphor identification and interpretation tasks across all the models, showing that the latter task deserves more research efforts in computational metaphor processing.

6.2 Breakdown analysis

We compare different models by open-class breakdowns in Table 2, finding that MetaPro 2.0 exceeds the baselines across all setups on both detection and interpretation tasks. Additionally, 36.88% and 36.11% of metaphoric MWEs can be correctly paraphrased by MetaPro2.0-STL-G and MetaPro2.0-MTL-G, respectively, exceeding BERT-G (STL: 28.46%; MTL: 26.21%), RoBERTa-G (STL:

Measure & Task		No. of candidates in a prompt			
		4	6	8	10
Train. min./ep.	STL	119.21	84.98	58.93	**46.22**
	MTL	95.92	75.43	52.21	**45.70**
Valid. min.	STL	118.19	49.32	38.23	**26.32**
	MTL	92.32	44.25	37.22	**23.20**
Valid. F1 score	Idnt.-STL	72.87	**73.29**	71.72	70.07
	Intp.-STL	**56.80**	55.90	56.00	55.92
	Idnt.-MTL	**73.04**	70.98	70.75	71.14
	Intp.-MTL	55.45	54.32	54.82	**55.78**

Table 3: MetaPro 2.0 hyperparameter analysis on the unit level. Metaphor interpretation uses gold metaphor identification labels. Min./ep. means minutes per epoch.

33.21%; MTL: 34.63%), GPT2-G (STL: 24.64%; MTL: 24.64%), and T5-G (STL: 28.93%; MTL: 26.07%). This supports our argued advantage of ALM that taking the semantics of the predicted positive and negative samples into account is helpful for the learning of MWEs.

6.3 Hyperparameter analysis

As shown in Table 3, including more candidates largely reduces training and inferring time costs. However, more candidates decrease F1 for metaphor identification and interpretation somewhat, because the longer prompt may impact the semantic representations of the original sentence. More candidates in a prompt also result in lengthy inputs. The longest input sequence of MetaPro 2.0 is 512 tokens after BPE. In practice, one can balance the sequence lengths and model utilities. Using an additional light classifier to recall more likely paraphrase candidates out of the full candidate list, then retrieving the best paraphrase via the main model may reduce inference time costs.

7 Conclusion

We proposed a novel PLM, ALM, for computational metaphor processing. We also created a large metaphor interpretation dataset (VMC-P) focusing on token-level metaphor paraphrases. By fine-tuning ALM and other parameter-size comparable PLMs with VMC-P, we found that fine-tuned ALM (MetaPro 2.0) outperforms the baselines in both metaphor identification and interpretation tasks.

According to the survey of Ge et al. (2023), extensive research works in metaphor processing focused on metaphor identification. Although it is a foundation task, the advancement of relevant tasks, e.g., linguistic and conceptual metaphor understandings have more practical values in downstream applications, e.g., enhancing machine understanding of metaphorical language (Mao et al., 2018, 2022), analyzing cognitive patterns of subjects through conceptual metaphor understanding (Han et al., 2022; Mao et al., 2023a, 2024b), and fostering language creativity (Yu and Wan, 2019; Stowe et al., 2021). Meeting the demands of these downstream applications necessitates the development of more functional metaphor processing systems beyond mere metaphor identification. Accurately paraphrasing metaphors plays a crucial role in bridging the gap between linguistic and conceptual metaphor interpretation, thereby providing an interpretable layer for text analysis (Wu et al., 2023, 2024; Cambria et al., 2024). We hope that the proposed ALM and VMC-P can inspire further research in this area and make a substantial contribution to the broader expansion of computational metaphor processing into downstream tasks.

Limitations

MetaPro 2.0 is a metaphor processing system, fine-tuned upon ALM and VMC-P dataset. It can identify and interpret English metaphors in a text preprocessing fashion. However, the current version does not support other languages besides English. Next, we did not consider metaphoric dependency between instances when we annotated the dataset (an instance is a short text with one or several sentences). The identification and interpretation of metaphors in one instance are independent of the metaphors from other instances. Thus, our system trained with the dataset is incapable of processing document-level dependent metaphors. Next, MetaPro 2.0 is a knowledge-based neurosymbolic system. If the paraphrase of a metaphor is out of the coverage of our developed knowledge base (the reference dictionary $\mathcal{D}$), MetaPro 2.0 cannot yield an accurate interpretation for it. Finally, MetaPro 2.0 cannot directly process very long texts, e.g., more than 512 tokens after BPE (the maximum of 512 tokens includes the tokens of the concatenated prompt). In practice, we will segment very long texts first, then feed them into our system.

Ethics Statement

This article follows the ACL Code of Ethics. The annotations are based on a public dataset that does not contain private data. The tool we developed is a data preprocessing technique for improving human and machine understanding of metaphorical language. To the best of our knowledge, there are no foreseeable potential risks of using this technique.

Acknowledgments

This research/project is supported by the Ministry of Education, Singapore under its MOE Academic Research Fund Tier 2 (STEM RIE2025 Award MOE-T2EP20123-0005) and by the RIE2025 Industry Alignment Fund – Industry Collaboration Projects (IAF-ICP) (Award I2301E0026), administered by A*STAR, as well as supported by Alibaba Group and NTU Singapore.

References

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*.

Yuri Bizzoni and Shalom Lappin. 2018. Predicting human metaphor paraphrase judgments with deep neural networks. In *Proceedings of the Workshop on Figurative Language Processing*, pages 45–55.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33:1877–1901.

Erik Cambria, Rui Mao, Melvin Chen, Zhaoxia Wang, and Seng-Beng Ho. 2023. Seven pillars for the future of artificial intelligence. *IEEE Intelligent Systems*, 38(6):62–69.

Erik Cambria, Xulang Zhang, Rui Mao, Melvin Chen, and Kenneth Kwok. 2024. SenticNet 8: Fusing emotion AI and commonsense AI for interpretable, trustworthy, and explainable affective computing. In *Proceedings of International Conference on Human-Computer Interaction (HCII)*.

Tuhin Chakrabarty, Arkadiy Saakyan, Debanjan Ghosh, and Smaranda Muresan. 2022. FLUTE: Figurative language understanding through textual explanations. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 7139–7159.

Xin Chen, Zhen Hai, Suge Wang, Deyu Li, Chao Wang, and Huanbo Luan. 2021. Metaphor identification: A contextual inconsistency based neural sequence labeling approach. *Neurocomputing*, 428:268–279.

Zhuang Chen and Tieyun Qian. 2020. Relation-aware collaborative learning for unified aspect-based sentiment analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3685–3694.

Minjin Choi, Sunkyung Lee, Eunseong Choi, Heesoo Park, Junhyuk Lee, Dongwon Lee, and Jongwuk Lee. 2021. MelBERT: Metaphor detection via contextualized late interaction using metaphorical identification theories. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1763–1773.

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 17th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186.

Kelvin Du, Frank Xing, Rui Mao, and Erik Cambria. 2024. An evaluation of reasoning capabilities of large language models in financial sentiment analysis. In *IEEE Conference on Artificial Intelligence (IEEE CAI)*, Singapore.

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. Bradford Books.

Mengshi Ge, Rui Mao, and Erik Cambria. 2022. Explainable metaphor identification inspired by conceptual metaphor theory. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(10):10681–10689.

Mengshi Ge, Rui Mao, and Erik Cambria. 2023. A survey on computational metaphor processing techniques: From identification, interpretation, generation to application. *Artificial Intelligence Review*, 56:1829–1895.

Sooji Han, Rui Mao, and Erik Cambria. 2022. Hierarchical attention network for explainable depression detection on Twitter aided by metaphor concept mappings. In *Proceedings of the 29th International Conference on Computational Linguistics (COLING)*, pages 94–104.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738.

Ilana Heintz, Ryan Gabbard, Mahesh Srinivasan, David Barner, Donald S Black, Marjorie Freedman, and Ralph Weischedel. 2013. Automatic extraction of linguistic metaphor with LDA topic modeling. In *Proceedings of the First Workshop on Metaphor in NLP*, pages 58–66.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength natural language processing in python.

Bipin Indurkhya. 2007. Creativity in interpreting poetic metaphors. *New directions in metaphor research*, pages 483–501.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Luuk Lagerwerf and Anoe Meijers. 2008. Openness in metaphorical and straightforward advertisements: Appreciation effects. *Journal of Advertising*, 37(2):19–30.

George Lakoff and Mark Johnson. 1980. *Metaphors We Live by*. University of Chicago Press.

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 Second Workshop on Figurative Language Processing*, pages 18–29.

Shuqun Li, Liang Yang, Weidong He, Shiqi Zhang, Jingjie Zeng, and Hongfei Lin. 2021. Label-enhanced hierarchical contextualized representation for sequential metaphor identification. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3533–3543.

Yucheng Li, Shun Wang, Chenghua Lin, and Frank Guerin. 2023a. Metaphor detection via explicit basic meanings modelling. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 91–100.

Yucheng Li, Shun Wang, Chenghua Lin, Frank Guerin, and Loïc Barrault. 2023b. FrameBERT: Conceptual metaphor detection with frame embedding learning. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1558–1563.

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. *arXiv e-prints*, pages arXiv–1907.

Rui Mao, Guanyi Chen, Xulang Zhang, Frank Guerin, and Erik Cambria. 2024a. GPTEval: A survey on assessments of ChatGPT and GPT-4. In *LREC-COLING*, pages 7844–7866.

Rui Mao, Kelvin Du, Yu Ma, Luyao Zhu, and Erik Cambria. 2023a. Discovering the cognition behind language: Financial metaphor analysis with MetaPro. In *2023 IEEE International Conference on Data Mining (ICDM)*, pages 1211–1216. IEEE.

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*, volume 35, pages 13534–13542.

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

Rui Mao, Xiao Li, Kai He, Mengshi Ge, and Erik Cambria. 2023b. MetaPro Online: A computational metaphor processing online system. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, volume 3, pages 127–135.

Rui Mao, Chenghua Lin, and Frank Guerin. 2018. Word embedding and WordNet based metaphor identification and interpretation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, volume 1.

Rui Mao, Qian Liu, Kai He, Wei Li, and Erik Cambria. 2023c. The biases of pre-trained language models: An empirical study on prompt-based sentiment analysis and emotion detection. *IEEE Transactions on Affective Computing*, 14(3):1743–1753.

Rui Mao, Tianwei Zhang, Qian Liu, Amir Hussain, and Erik Cambria. 2024b. Unveiling diplomatic narratives: Analyzing United Nations Security Council debates through metaphorical cognition. In *Proceedings of the Annual Meeting of the Cognitive Science Society (CogSci)*.

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*.

National Academies of Sciences, Engineering, and Medicine and others. 2018. *How people learn II: Learners, contexts, and cultures*. National Academies Press.

Roberto Navigli and Simone Paolo Ponzetto. 2012. BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250.

Joakim Nivre, Marie-Catherine De Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, et al. 2016. Universal dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation*, pages 1659–1666.

OpenAI. 2023. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 27–35.

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

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. 2018. Improving language understanding by generative pre-training. *OpenAI*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Ekaterina Shutova. 2015. Design and evaluation of metaphor processing systems. *Computational Linguistics*, 41(4):579–623.

Gerard J Steen, Aletta G Dorst, J Berenike Herrmann, Anna Kaal, Tina Krennmayr, and Trijntje Pasma. 2010. *A method for linguistic metaphor identification: From MIP to MIPVU*, volume 14. John Benjamins Publishing.

Kevin Stowe, Tuhin Chakrabarty, Nanyun Peng, Smaranda Muresan, and Iryna Gurevych. 2021. Metaphor generation with conceptual mappings. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pages 6724–6736.

Kevin Stowe, Sarah Moeller, Laura Michaelis, and Martha Palmer. 2019. Linguistic analysis improves neural metaphor detection. In *Proceedings of the 23rd Conference on Computational Natural Language Learning*, pages 362–371.

Kevin Stowe, Prasetya Utama, and Iryna Gurevych. 2022. IMPLI: Investigating NLI models' performance on figurative language. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pages 5375–5388.

Chang Su, Ying Peng, Shuman Huang, and Yijiang Chen. 2020. A metaphor comprehension method based on culture-related hierarchical semantic model. *Neural Processing Letters*, 51(3):2807–2826.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

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.

Xiaobao Wu, Xinshuai Dong, Thong Nguyen, and Anh Tuan Luu. 2023. Effective neural topic modeling with embedding clustering regularization. In *International Conference on Machine Learning*.

Xiaobao Wu, Fengjun Pan, Thong Nguyen, Yichao Feng, Chaoqun Liu, Cong-Duy Nguyen, and Anh Tuan Luu. 2024. On the affinity, rationality, and diversity of hierarchical topic modeling. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Bo Xu, Tingting Li, Junzhe Zheng, Mehdi Naseriparsa, Zhehuan Zhao, Hongfei Lin, and Feng Xia. 2022. MET-Meme: A multimodal meme dataset rich in metaphors. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2887–2899.

Zhiwei Yu and Xiaojun Wan. 2019. How to avoid sentences spelling boring? towards a neural approach to unsupervised metaphor generation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 861–871.

Omnia Zayed, John Philip McCrae, and Paul Buitelaar. 2020. Figure me out: a gold standard dataset for metaphor interpretation. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 5810–5819.

A Rationality of the Annotation Method

Metaphor interpretation is a highly subjective and creative process, influenced by individuals' intuitions and personal experiences (Indurkhya, 2007). Despite this, there is a lack of theoretical research providing practical guidance for humans to paraphrase metaphors with minimal divergence. Consequently, achieving independent metaphor paraphrase annotations with high-level agreement is challenging, as there can be multiple valid interpretations for the same metaphor. For instance, the metaphorical expression "she *devoured* his novels", could be paraphrased as "she avidly enjoyed his novels" or "she eagerly read his novels".

Considering the purpose of developing VMC-P dataset is to train E2E metaphor processing systems that can be used in real-world applications (as we argued at the beginning of Section 3), our annotation method involves an annotate-then-evaluate process. Initially, an expert annotates metaphor paraphrases, which are then evaluated by two independent raters to determine their acceptability.

This annotation process is rational for several reasons: (1) An E2E metaphor processing system trained on our VMC-P dataset is supposed to produce human-acceptable metaphor paraphrases, making the dataset unnecessary to exhaustively gather all possible paraphrases of the same metaphor from human annotators. (2) Due to the subjective nature of metaphor interpretation, obtaining majority-agreed paraphrases from independent annotations is challenging, as the selected annotators may not represent the preference of a wider population. Different annotation groups likely produce varying majority-agreed paraphrases in this domain. (3) Utilizing an expert with profound linguistic knowledge and responsibility to annotate metaphor paraphrases can ensure the annotation accuracy. A single expert annotator can provide more consistent annotations across various tasks compared to distributing the tasks among multiple annotators in a crowd-sourcing approach. Consistent annotations are particularly beneficial for machine learning models in understanding data patterns. (4) The high agreement rate among the two independent raters (see Section 3.2) can further validate the expert's annotations, ensuring the dataset's quality.

Therefore, the annotate-then-evaluate process enhances the utility (e.g., obtaining more human-annotated data within specific cost constraints) and reliability (scientific validation by external raters) of the developed VMC-P dataset for training E2E metaphor processing systems.

Metaphor Identification Procedure:

1. Read the entire text–discourse to establish a general understanding of the meaning.
2. Determine the lexical units in the text–discourse.
3. (a) For each lexical unit in the text, establish its meaning in context, that is, how it applies to an entity, relation, or attribute in the situation evoked by the text (contextual meaning). Take into account what comes before and after the lexical unit.
   (b) For each lexical unit, determine if it has a more basic contemporary meaning in other contexts than the one in the given context. For our purposes, basic meanings tend to be
   - More concrete; what they evoke is easier to imagine, see, hear, feel, smell, and taste.
   - Related to bodily action.
   - More precise (as opposed to vague).
   - Historically older.

   Basic meanings are not necessarily the most frequent meanings of the lexical unit.
   (c) If the lexical unit has a more basic current–contemporary meaning in other contexts than the given context, decide whether the contextual meaning contrasts with the basic meaning but can be understood in comparison with it.
4. If yes, mark the lexical unit as metaphorical.

Figure 3: Metaphor Identification Procedure.

B Metaphor Identification Procedure

The content (Metaphor Identification Procedure, MIP) in Figure 3 is from the work of Praggle-jaz (2007), which is part of the material we used to train the annotator and raters for the metaphor interpretation annotation task. It is also one of the linguistic foundations that inspired the design of our anomalous language modeling-based pre-training paradigm and the following fine-tuning. Specifically, the contextual meaning of an original word/a metaphor contrasts with the basic meaning of the anomalous lexical substitution/metaphor in pre-training/fine-tuning. Thus, (1) both anomalous words and metaphors are distinguishable from context words; (2) the contextual meanings of anomalous words/metaphors should be closer to the meanings of original words/literal counterparts of the metaphors and further from other candidates that represent different meanings to make sense of a

sentence in semantic space.

C Dataset Information


```
{‘ID’: ‘trn_976’,
‘doc_ID’: ‘ac2-fragment06’,
‘sent_ID’: ‘1465’,
‘sent’: “‘ Do n’t they realise they ’re playing with political dynamite ?”,
‘metaphor_index_list’: [[7, 8], [10]],
‘pos_list’: [‘underestimating’, ‘risks’],
‘neg_list’: [[‘exploiting’, ‘trifling’, ..., ‘taking on’, ‘making for’], [‘explosive compound’, ‘explode’]],
‘lemma’: “‘ do not they realise they be play with political dynamite ?”,
‘pos_tags’: [“‘’, ‘VB’, ‘RB’, ‘PRP’, ‘VB’, ‘PRP’, ‘VBP’, ‘VBG’, ‘IN’, ‘JJ’, ‘NN’, ‘.’],
‘open_class’: [‘O’, ‘VERB’, ‘ADV’, ‘O’, ‘VERB’, ‘O’, ‘VERB’, ‘VERB’, ‘O’, ‘ADJ’, ‘NOUN’, ‘O’],
‘genre’: ‘fiction’}
```


Figure 4: An example of data in our VMC-P dataset.

In line with VMC dataset[11], our VMC-P dataset is also licensed under a Creative Commons Attribution-ShareAlike 3.0 Unported License[12]. An example of data in our dataset can be viewed in Figure 4. ‘ID’ is the index of an instance in our dataset. The ‘doc_ID’ and ‘sent_ID’ are inherited from the original VMC dataset. ‘sent’ is the original sentence with tokenization. ‘metaphor_index_list’ denotes the position of a metaphor. If a sub-list has more than 1 element, it means the words corresponding to the indices within the sub-list are metaphoric MWEs, e.g., “play with” at Indices 7 and 8, respectively. ‘pos_list’ denotes the gold paraphrases. ‘neg_list’ denotes negative samples. The elements of ‘metaphor_index_list’, ‘pos_list’, and ‘neg_list’ are aligned by indices. ‘lemma’ and ‘pos_tags’ were generated by spaCy ‘en_core_web_sm’ (Honnibal et al., 2020), indicating the lemmatized sequence and the PoS sequence, respectively. The PoS tags follow the Universal Dependencies scheme (Nivre et al., 2016). ‘open_class’ denotes if a token at the same position is a verb, a noun, an adjective, an adverb, or others (O). In order to ensure that future evaluations can be maintained at the same statistical standard, we tokenize the original sentences and include the PoS labels. ‘genre’ denotes the genre of the text, which is inherited from the original VMC dataset, including academic texts, news, fiction and conversation.

The distributions of sequence lengths can be viewed in Figure 5. The detailed statistics of our VMC-P dataset can be viewed in Table 4.

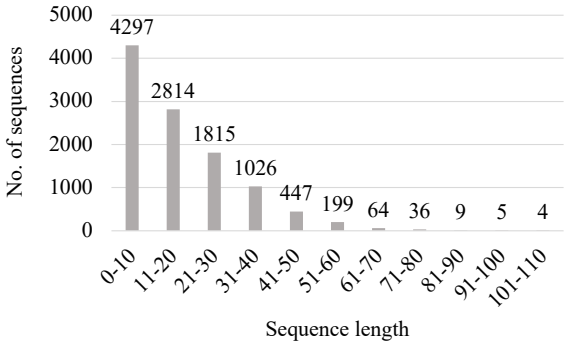


Figure 5: The distributions of sequence lengths.

	Train	Valid	Test	All
# seq	6,653	2,063	2,000	10,716
Min len	1	1	1	1
Max len	110	86	107	110
Avg len	17	15	23	18
% LS	59.10%	56.62%	12.25%	49.88%
% news	15.84%	14.69%	37.40%	19.64%
% fictions	20.94%	31.60%	29.85%	24.65%
% academic	22.34%	14.01%	18.80%	20.07%
% conversation	40.88%	39.70%	13.95%	35.63%
# tokens	111,718	30,170	46,129	188,017
% LT	94.67%	94.07%	91.02%	93.68%
% MT	5.33%	5.93%	8.98%	6.32%
# MU	5,950	1,789	4,141	11,880
% VB	44.22%	44.77%	43.71%	44.12%
% NN	34.30%	31.64%	34.15%	33.85%
% ADJ	14.52%	15.71%	14.75%	14.78%
% ADV	2.25%	2.57%	2.49%	2.38%
% MWE	4.64%	5.31%	4.71%	4.76%
% other	0.07%	0.00%	0.19%	0.10%
Avg # C/MU	31.47	32.30	30.23	31.16

Table 4: Dataset statistics. # seq denotes the number of sequences. Min len denotes the minimum length of the sequences, while max len and avg len denote the maximum and the average sequence lengths, respectively. % LS denotes the percentage of literal sequences among all sequences. # tokens denotes the number of tokens. % LT denotes the percentage of literal tokens among all tokens, while % MT denotes the percentage of metaphoric tokens. # MU denotes the number of metaphoric units. A metaphor unit is a metaphoric SWE or a metaphoric MWE. % VB denotes the percentage of verbs among all metaphoric units. NN, ADJ, ADV, and MWE denote nouns, adjectives, adverbs, and multi-word expressions, respectively. Avg # C/MU denotes the average number of candidates (positive and negative samples) per metaphoric unit.

[11] https://www.vismet.org/metcor/license.html

[12] https://creativecommons.org/licenses/by-sa/3.0/

D Evaluation Metrics

We use F1 scores as the main metric for metaphor identification and interpretation overall evaluation.

$$F1 = \frac{2 \times precision \times recall}{precision + recall},$$

$$precision = \frac{TP}{TP + FP},$$

$$recall = \frac{TP}{TP + FN}.$$

The metaphor identification overall evaluation is evaluated on the unit level. True Positive (TP) is defined as a predicted metaphoric unit whose ground truth label is also metaphoric. A metaphor unit is a metaphoric SWE or MWE. A metaphoric MWE prediction is correct only if all the B and I labels of it are correctly predicted. False Positive (FP) is defined as a predicted metaphoric unit whose ground truth label is literal. False Negative (FN) is defined as a predicted literal unit whose ground truth label is metaphoric. In metaphor identification breakdown analysis, we evaluate the performance of different open-class PoS on the token level (the tokens in 'sent' in the VMC-P dataset), because we did not annotate literal MWEs. We use accuracy as the breakdown evaluation metric. The accuracy of a PoS breakdown of metaphor identification is defined as the number of correct predictions on the token level above the total number of tokens within the PoS. A metaphor identification prediction is correct if the predicted label is the same as the ground truth label on the token level.

The metaphor interpretation overall evaluation is evaluated on the unit level. TP is defined as both the metaphor identification and paraphrase predictions are the same as the ground truth metaphoric identification and paraphrase labels. FP is defined as a predicted metaphoric unit whose ground truth label is literal, and a predicted metaphoric unit whose ground truth label is metaphoric, while the predicted paraphrase of the identified metaphoric unit is different from the ground truth paraphrase. FN is defined as a predicted literal unit whose ground truth label is metaphoric. In metaphor interpretation breakdown analysis, we evaluate the performance of different open-class PoS on the token level and use accuracy as the breakdown evaluation metric. We use ground truth metaphor identification labels for metaphor interpretation breakdown analysis. The accuracy of a PoS breakdown of metaphor interpretation is defined as the number of correct paraphrased tokens above the total number of tokens that should be paraphrased within the PoS. A metaphor interpretation prediction is correct if the predicted paraphrase is the exact same as the ground truth paraphrase. If the predicted paraphrase of an MWE is correct, all words within the MWE are considered as correctly paraphrased.

Our evaluation method is conservative because we consider a prediction is correct, only if it is the exact same as its gold label. We do not use ranking-based metrics, e.g., Hits @ K, because a metaphor processing system that only yields a paraphrased sequence for an input sequence is more supportive in downstream tasks, compared to a system that yields several candidate paraphrases. Besides, we cannot evaluate the quality of other high-ranking predictions by our dataset, although a metaphor can have multi-paraphrases in real-world texts.

E LLM performance on VMC-P

Currently, a wide range of LLM evaluations have been conducted across various domains (Mao et al., 2024a). To the best of our knowledge, the assessments focusing on metaphor identification and interpretation by LLMs have not yet been thoroughly explored. This gap exists due to the lack of extensive annotated datasets that encompass both the identification and interpretation of metaphors. Thus, we demonstrate the preliminary performance of GPT-3.5-turbo and GPT-4 (OpenAI, 2023) on our developed VMC-P testing set. While the results of GPTs and ALM are presented together in Table 5, it does not mean that the two types of models are objectively comparable for the following reasons: ***a***) GPT-3.5-turbo and GPT-4 were not explicitly fine-tuned for computational metaphor processing tasks; ***b***) The parameter sizes of GPT-3.5-turbo (175 billion) and GPT-4 (1.76 trillion) are much larger than ALM (355 million); ***c***) GPT-3.5-turbo and GPT-4, designed for generation, and MetaPro 2.0, developed for E2E metaphor interpretation, serve distinct purposes and entail varying deployment and inference costs. The inclusion of results from these models in the same table serves the purpose of illustrating the disparity between a small expert system (MetaPro 2.0) and robust generative AI (GPT-3.5-turbo and GPT-4) in terms of their strengths and weaknesses on our newly developed VMC-P dataset.

We used OpenAI API to access GPT-3.5-turbo (model name: gpt-3.5-turbo; temperature: 0; top_p:

1) and GPT-4 (model name: gpt-4; temperature: 0; top_p: 1) on 8th February 2024. There were two prompts employed in our large GPT-based metaphor identification evaluation, namely direct prompting (GPT_{DP}) and Chain-of-Thought prompting (GPT_{COT}). The direct prompt is designed as follows:

```
You are a linguistic expert in metaphor analysis. Given the text below, the task is to identify metaphors from the text. Please tokenize the textual string by whitespace and return the identified metaphorical tokens only. If there is no metaphor in the text, return NA. Use concise language to present the result.
Text: [a testing instance]
```

The text is a testing instance from VMC-P (e.g., joining tokens in the 'sent' list of Figure 4 with whitespaces). In order to augment the logical reasoning capabilities of the large GPTs, we incorporate a Chain-of-Thought methodology (Wei et al., 2022) into our prompts. Recognizing the theoretical robustness of MIP (Pragglejaz, 2007) in guiding human annotation of metaphors, we concatenate the direct prompts with the content of MIP (refer to Figure 3) as the Chain-of-Thought prompt.

For evaluating large GPTs in the metaphor interpretation task, the prompt is designed as follows:

```
You are a linguistic expert in metaphor analysis. Given the text below and a candidate list, which one is the best literal counterpart for the metaphor "[a metaphor]" among the candidate list? Only return a choice with concise language in the answer. Do not explain why.
Text: [a testing instance]
Candidates: [the combination of a gold paraphrase and the corresponding negative samples]
```

The text is also a testing instance from the proposed VMC-P dataset. Each evaluation iteration only focuses on a target metaphor (a gold identification label-based, or predicted identification label-based metaphor). The candidates are the combination of the gold paraphrase of the target metaphor and its corresponding negative samples. The position of the gold paraphrase is randomized among the candidates. There are also two evaluation tasks, namely gold metaphor identification label-based, and predicted metaphor identification label-based paraphrase evaluation. The metaphor identification predictions are from the former prompting models, e.g., GPT_{DP} and GPT_{COT}.

In Table 5, we can observe that both GPT-3.5-turbo and GPT-4 yield very weak performance in the metaphor identification task under the direct prompting and Chain-of-Thought prompting setups. This is because of their very frequent false positive errors. The large GPTs encounter difficulty in identifying fine-grained metaphorical words, because many large GPT-identified metaphors are long phrases. Such coarse-grained metaphor identification predictions do not benefit conceptual metaphor interpretation, because it would be difficult and ambiguous to abstract target and source concepts from long phrases with complicated syntactic structures (see Conceptual Metaphor Theory (Lakoff and Johnson, 1980) for the relevant concepts and examples). On the other hand, the GPTs also exhibit shortcomings in identifying metaphors, such as false negative errors, which underscores their limited proficiency in professional linguistic analysis of metaphorical language.

Task	Model	P	R	F1
Identification	$\text{GPT-3.5}_{\text{DP}}$	14.98	12.10	13.39
	$\text{GPT-3.5}_{\text{COT}}$	19.44	6.33	9.55
	GPT-4_{DP}	13.92	7.44	9.69
	$\text{GPT-4}_{\text{COT}}$	32.18	27.34	29.56
	$\text{MetaPro2.0}_{\text{STL}}$	**72.41**	68.25	**70.23***
	$\text{MetaPro2.0}_{\text{MTL}}$	70.28	**70.17**	70.20*
Interpretation (gold idnt. label -based)	GPT-3.5-G	**36.32**	100	**53.29**
	GPT-4-G	36.15	100	53.10
	$\text{MetaPro2.0-G}_{\text{STL}}$	30.28	100	46.46
	$\text{MetaPro2.0-G}_{\text{MTL}}$	29.02	100	44.97
Interpretation (predicted idnt. label-based)	$\text{GPT-3.5-P}_{\text{DP}}$	4.23	3.40	3.77
	$\text{GPT-3.5-P}_{\text{COT}}$	6.16	2.00	3.02
	$\text{GPT-4-P}_{\text{DP}}$	4.34	2.32	3.02
	$\text{GPT-4-P}_{\text{COT}}$	9.84	8.36	9.04
	$\text{MetaPro2.0-P}_{\text{STL}}$	**27.63**	**69.73**	**39.55***
	$\text{MetaPro2.0-P}_{\text{MTL}}$	26.49	69.11	38.28*

Table 5: GPT-3.5-turbo and GPT-4 performance on the VMC-P testing set. COT denotes Chain-of-Thought, based on MIP. * denotes the improvement is statistically significant ($p < 0.001$) on a two-tailed test.

With MIP-informed Chain-of-Thought prompting, $\text{GPT-4}_{\text{COT}}$ yields significant improvements in the evaluation metrics, compared to other large GPT models. This suggests that professional linguistic knowledge, e.g., MIP, can enhance the performance of GPT-4 in accurately predicting metaphor identification tasks, although this improvement is not observed in the earlier version, GPT-3.5-turbo. This phenomenon raises a practical question regarding the use of prompt-based LLMs for computational linguistics tasks resembling metaphor identification: How can the effectiveness of prompts, originally devised for earlier or limited LLMs, be preserved across different or future versions? Examples where prompts succeeded in earlier versions of LLMs but failed in later versions can also be observed in financial sentiment

Task	Labels
Aspect Extraction	B: the beginning word of an aspect unit; I: the inside word of an aspect unit; O: the outside word of an aspect unit.
Opinion Extraction	B: the beginning word of an opinion unit; I: the inside word of an opinion unit; O: the outside word of an opinion unit.
Aspect -based Sentiment Analysis	Background: A background word without sentiment; Positive: an aspect word with positive sentiment; Negative: an aspect word with negative sentiment; Neutral: an aspect word with neutral sentiment; Conflict: an aspect word with conflict sentiment.
Named Entity Recognition	O, B-PER, I-PER, B-ORG, I-ORG, B-LOC, I-LOC, B-MISC, I-MISC, where PER, ORG, LOC and MICS denote persons, organizations, locations and miscellaneous names, respectively. B, I, O denote the beginning, inside and outside of a named entity.

Table 6: The label sets of aspect extraction, opinion extraction, aspect-based sentiment analysis and named entity recognition tasks.

analysis tasks (Du et al., 2024). Determining a robust and effective prompt for a specific task is challenging due to the heuristic nature of many current approaches to prompt engineering (Mao et al., 2023c) and the unpredictable preferences of fast-evolved LLMs in the future. The prompting approaches, which include considerations such as phrasing, linguistic structures, as well as prompting knowledge and logic, lack a systematic methodology for identifying the optimal and robust prompt for different foundation models.

Finally, GPT-3.5/4-G outperforms MetaPro 2.0-G in interpreting metaphors when utilizing gold metaphor identification labels. This outcome is unsurprising due to the extensive pre-training data and larger parameter size of GPT-4. However, in pursuit of achieving E2E metaphor interpretation (the main goal of this work), GPT-3.5/4-P_{DP} and GPT-3.5/4-P_{COT} lag behind MetaPro 2.0-P in metaphor interpretation tasks, based on predicted metaphor identification labels. Considering the costs of employing large GPTs and the challenges in parsing structured data from their generated text[13], MetaPro 2.0 retains advantages in E2E metaphor interpretation tasks in real-world application scenarios. The E2E processing gaps between the GPTs and MetaPro 2.0 also highlight the value of VMC-P for training expert systems in the domain of computational metaphor processing, because the relevant linguistic knowledge can not be easily learned from general corpora by pre-training.

[13]Although we have instructed GPT-3.5 and GPT-4 to produce only the required results, it is inevitable that the outputs may exhibit varied linguistic structures and include extraneous contextual information. Thus, it is difficult to parse the desired predictions, e.g., metaphor tokens, and the interpretation of the specific metaphor tokens from GPT-3.5/4 generated text.

Task	Model	Micro F1	Acc
Aspect Extraction	BERT	72.05	68.63
	RoBERTa	89.76	88.37
	GPT2	67.22	64.81
	T5	89.10	88.15
	ALM	**90.59**	**89.87**
Opinion Extraction	BERT	78.26	75.94
	RoBERTa	85.13	87.81
	GPT2	71.42	69.08
	T5	81.48	86.28
	ALM	**86.42**	**88.77**
Aspect-based Sentiment Analysis	BERT	60.11	59.32
	RoBERTa	79.36	77.11
	GPT2	53.05	52.09
	T5	73.89	73.06
	ALM	**80.36**	**78.70**
Named Entity Recognition	BERT	79.38	77.30
	RoBERTa	90.81	92.38
	GPT2	65.93	64.70
	T5	91.33	92.56
	ALM	**92.04**	**93.29**
Average	BERT	72.45	70.30
	RoBERTa	86.26	86.42
	GPT2	64.40	62.67
	T5	83.95	85.01
	ALM	**87.35**	**87.66**

Table 7: ALM and other PLM benchmarking results on aspect extraction, opinion extraction, aspect-based sentiment analysis and named entity recognition tasks, averaged over 5 runs.

F Testing ALM on Other Tasks

We aim to propose useful resources, e.g., a dataset and a PLM for the learning of E2E metaphor interpretation in this work. However, since sequence tagging (e.g., detecting anomalous words from a sequence) is one of the pre-training tasks of ALM, we also test its performance on other sequence tagging tasks such as aspect extraction, opinion extraction, aspect-based sentiment analysis, and named entity recognition. The first three tasks are evaluated with the restaurant dataset from Pontiki et al. (2014), following the training, validation, and testing set splits of Chen and Qian (2020). Named entity recognition is evaluated with the CoNLL 2003 NER dataset from Tjong Kim Sang and De Meulder (2003), following the splits of huggingface datasets[14]. The label set of each task can be viewed in Table 6. As seen in the table, different tasks have different label sets and annotation paradigms, indicating the fact that the learning of those tasks needs to model different dependency relationships between words within a textual sequence.

We test the same baselines that are introduced in Section 5 on the sequence tagging tasks. As seen in Table 7, our metaphor processing-tailored PLM also has advantages in processing those non-metaphor tasks because ALM exceeds them across all the tasks. It shows that a PLM can also gain gen-

[14]https://https://huggingface.co/datasets/conll2003

eral knowledge by the ALM pre-training paradigm.

While the gains in non-metaphor tasks (+1.09% F1 on average) are not as substantial as those observed in sequential metaphor identification (+3.20% F1 on average), it is important to consider the holistic utility of our PLM – excelling in metaphor processing while maintaining competitive performance in other tasks. Besides, it also shows that a task-oriented PLM can achieve better pre-training utilities than general PLMs on specific tasks. Probably, the "divide-and-conquer" strategy, e.g., using different pre-training paradigms to handle very different reasoning tasks, may be also applicable to LLMs in enhancing their performance in different domains (Cambria et al., 2023).