

Enhancing Large Language Models with Multimodality and Knowledge Graphs for Hallucination-free Open-set Object Recognition

Xinfu Liu
Hohai University
Nanjing, China
liuxinfu@hhu.edu.cn

Yirui Wu
Hohai University
Nanjing, China
wuyirui@hhu.edu.cn

Yuting Zhou
Hohai University
Nanjing, China
zhouyuting@hhu.edu.cn

Junyang Chen
Shenzhen University
Shenzhen, China
junyangchen@szu.edu.cn

Huan Wang
Huazhong Agricultural University
Wuhan, China
hwang@mail.hzau.edu.cn

Ye Liu
Nanjing Agricultural University
Nanjing, China
yeliu@njau.edu.cn

Shaohua Wan
University of Electronic Science and
Technology of China
Shenzhen, China
shaohua.wan@uestc.edu.cn

ABSTRACT

Open-set object recognition plays a significant role in today’s production and daily life, such as in surface defect detection, biometric identification, and autonomous driving recognition. However, due to the diversity of unknown categories and the complexity of scenarios, existing methods often perform poorly. Therefore, open-set object recognition remains an important and popular research topic. Recently, collaborative utilization of multiple pre-trained Large Language Models (LLMs) has emerged rapidly, which becomes a new research hotspot in addressing open-set object recognition task. Among this, a core challenge lies in amplifying the strengths of individual LLM while mitigating their weaknesses. In this paper, we propose a novel joint framework tailored for open-set object recognition tasks, aiming to more efficiently harness the capabilities of diverse LLMs and Knowledge Graphs(KGs). Initially, for the text data generated by textual LLMs, we use Wikipedia to correct and complete it. Then, we designed a text-image multi-modal fusion method to further correct and complete the text information by utilizing the implicit semantic information in the image. Additionally, we propose some novel designs to alleviate the hallucination issue of LLMs and reduce their instability. Extensive experiments demonstrate that our approach outperforms all the comparison methods.

VLDB Workshop Reference Format:

Xinfu Liu, Yirui Wu, Yuting Zhou, Junyang Chen, Huan Wang, Ye Liu, and Shaohua Wan. Enhancing Large Language Models with Multimodality and Knowledge Graphs for Hallucination-free Open-set Object Recognition. VLDB 2024 Workshop: LLM+KG.

This work is licensed under the Creative Commons BY-NC-ND 4.0 International License. Visit <https://creativecommons.org/licenses/by-nc-nd/4.0/> to view a copy of this license. For any use beyond those covered by this license, obtain permission by emailing info@vldb.org. Copyright is held by the owner/author(s). Publication rights

VLDB Workshop Artifact Availability:

The source code, data, and/or other artifacts have been made available at <https://github.com/Fire-Liu001/LLMKG>.

1 INTRODUCTION

Open-set object recognition aims to identify if an object is from a closed-set class that has appeared during training, or an open-set class that has not been encountered in the training set[49]. It is commonly encountered in daily life and represents a fundamental yet critical task. Its applications span various domains including autonomous driving[24], malware classification[7], medical image analysis[50], and surface defect detection[31], etc. The failure of open-set object recognition tasks can lead to severe consequences. Therefore, the academic community has continuously engaged and proposed numerous excellent methodologies[14, 22, 46, 55].

For open-set object recognition tasks, the core challenge lies in the selection of discriminate features while mitigating the effects of spurious-discriminative features. Spurious-discriminative features refer to perform well within closed-set classes but may cause confusion between open-set classes and closed-set classes[49]. Previous researches[45, 48] have suggested that a good open-set object recognition network minimizes reliance on spurious-discriminative features, thereby enabling more discriminative features and consequently enhancing recognition accuracy. For Example, as shown in Fig 1, the feature "with bill" belongs to a spurious-discriminative characteristic, which should be avoided as much as possible. To address this issue, researchers have made numerous attempts. Among them, a typical approach is to incorporate additional virtual open-set classes to aid in determining the attribution of original categories[21, 22, 45]. These methods have achieved certain effects, but due to

licensed to the VLDB Endowment.
Proceedings of the VLDB Endowment. ISSN 2150-8097.

the vast diversity of the real world, open-set object recognition still poses significant challenges. Moreover, numerous approaches re-train the dataset to enable the classifier to adapt to new categories[34]. However, due to privacy concerns or other various reasons, some data becomes inaccessible, rendering the retraining process challenging.

Recently, research on Large Language Models(LLMs) has become a hot topic, and LLMs have also been utilized to tackle various downstream tasks[16, 17, 51]. Furthermore, by combining multiple LLMs to leverage their respective strengths and compensate for their weaknesses, remarkable results have been achieved in handling downstream tasks collaboratively[49, 62, 64]. This paper inspired by LMC[49], our method also utilizes the collaborative manner of LLMs, making full use of the respective advantages of ChatGPT[6], CLIP[1], DALL-E[38], and DINO[61] to tackle the open-set object recognition task. However, despite the excellent performance of LLMs, they also possess some inherent drawbacks, such as the issue of factual hallucinations. Factual hallucinations can be understood as the appearance of seemingly genuine yet incorrect or distorted facts during the perception or generation of information. In the context of LLMs or Artificial Intelligence(AI), this typically refers to the production of outputs by the model that do not align with factual reality[26]. Existing research [5, 11, 40, 56] indicates that the main limitation of LLMs lies in their reliance solely on massive training datasets for knowledge reservoirs, without the capability to communicate with the external world in real-time.

As a crucial part of the knowledge representation system, Knowledge Graphs (KGs) can store a vast amount of knowledge that is closely related to the real world. Through rigorous reasoning, these pieces of knowledge ensure the coherence and causality of information. Therefore, the application of KGs is widespread in the field of knowledge modeling. Surprisingly, the internal knowledge of KGs and LLMs can complement each other, making the expression and utilization of knowledge more complete[59]. Hence, leveraging KGs to refine the incomplete outputs of LLMs presents an excellent solution. To address this issue, some methods[4, 13, 26] have introduced Knowledge Graphs(KGs) and utilized the additional knowledge within them to correct the erroneous or incomplete responses of LLMs.

However, these methods are solely applied within a single LLM and have not been explored under the condition of multiple LLMs collaborating or in a multimodal context. Moreover, they have not addressed the issue of factual hallucinations for open-set object recognition tasks.

In this paper, regarding the open-set object recognition task, we propose an optimized collaborative LLMs method, leveraging the unique strengths of various LLMs to fully exploit their advantages under multi-modal conditions. Additionally, we have introduced KGs to mitigate the factual hallucination issue originated from LLMs. Contributions of this paper can be summarized as follows:

- We propose an enhanced collaborative LLMs method, incorporating KGs to alleviate the prevalent factual hallucination issue in LLMs.
- We propose an image-text knowledge fusion method that elevates the performance of open-set object recognition under multi-modal conditions.

- Comprehensive experiments demonstrate that our approach outperforms all the comparison methods.

The remainder of this paper is organized as follows: Section 2 introduces the related work, including the latest developments in open-set object recognition methods, the issue of textual hallucinations in LLMs, and the progress of methods enhancing pre-trained large models with Knowledge Graphs. Section 3 presents the details of our proposed method, encompassing the LLMs iterative cycling, text-enhancement module, and text-image knowledge fusion module. Subsequently, we conduct extensive experiments in Section 4, including comparison experiments with other methods and ablation studies on various modules of our approach. Finally, we summarize the work of the entire paper.

2 RELATED WORKS

2.1 Open-set Object Recognition

Open-set object recognition, a pivotal concept in computer vision and machine learning, refers to the capability of a model to not only discern known object categories present in the training set but also identify instances that do not belong to any of the predefined classes, i.e., objects of unknown categories[60]. This recognition prowess endows the model with enhanced robustness and flexibility when confronted with the intricacies of real-world scenarios. Common strategies for open-set recognition encompass quantifying prediction uncertainty[52], modifying the softmax layer[46], and integrating generative and discriminative models[55]. The quantification of prediction uncertainty often leverages entropy or softmax scores to identify instances where the model exhibits a lack of confidence or encounters unfamiliar patterns. Adjustments to the softmax layer, exemplified by the OpenMax algorithm, involve refining the layer through extreme value theory to analyze distance distributions, thereby mitigating misclassifications of unknown classes.

The integration of generative and discriminative models for open-set recognition is multifaceted. On the generative side, data generation techniques can produce synthetic images of hypothetical unknown classes to augment model training. Conversely, discriminative approaches, leveraging techniques such as clustering or deep neural networks, discern between known and unknown categories. Open-set object recognition holds immense potential for application across diverse domains. In autonomous vehicle systems, for instance, it is crucial for vehicles to detect a myriad of known and unknown obstacles on the road. Similarly, in intelligent surveillance systems, cameras must differentiate between identified individuals and those not previously encountered. Furthermore, in medical image analysis, models must be capable of recognizing both known disease types and potential novel pathologies.

There have been numerous studies on open-set object recognition task[15, 52, 60], and it remains a hot research issue. Initially, Bendale et al.[9] first attempted to utilize deep neural networks for the task of open-set object recognition. Subsequently, Ge et al.[25] presented a conceptually new and flexible method for multi-class open-set classification. Unlike previous methods, their method is able to provide explicit modelling and decision score for unknown classes. After that, Neal et al.[46] introduced a dataset augmentation technique that being called counterfactual image generation, which



Figure 1: Description of spurious-discriminative features. As can be seen from the figure, it is easy to distinguish Woodpecker, Teddy dog and Persian cat by using the feature of "with bill". However, it is difficult to distinguish between Bald Eagle and Black Stork. Among them, DIAS is a recently published open-set object recognition method.

based on generative adversarial networks, generates examples that are close to training set examples yet do not belong to any training category. Later on, Kong et al.[35] proposed OpenGAN, which addresses the limitation of each approach by combining them with several technical insights. Then, Vaze et al.[55] first demonstrated that the ability of a classifier to make the "none-of-above" decision is highly correlated with its accuracy on the closed-set classes. Besides, Esmaeilpour et al.[22] studied the problem of zero-shot out-of-distribution (OOD) detection, which still performs the same two tasks in testing but has no training except using the given known class names, and then proposed a novel method (called ZOC) to solve the problem. Recently, Fu et al.[23] hold a opinion that current open set recognition techniques mainly concentrate on constructing decision boundaries rooted in holistic feature representations, demonstrating proficiency across broad-category image datasets. Nevertheless, when dealing with fine-grained image collections, where objects exhibit remarkable overall similarity, differentiating between known and novel classes solely based on these holistic features becomes challenging. To tackle this limitation, they introduce the Progressive Learning Vision Transformer (PLViT), integrating a coarse-to-fine refinement approach. This approach dynamically combines and optimizes both holistic and localized feature representations within an angular framework, thereby enhancing the discrimination capability of decision boundaries.

Most relevant to our work, Qu et al.[49] proposed a novel framework named LMC to tackle the open-set object recognition challenge via collaborating different off-the-shelf large models in a training-free manner. Based on LMC, we introduce KGs and multi-modal LLMs, which perfected the LMC method, and achieved satisfactory results.

2.2 Hallucination in LLMs

Within the context of LLMs, hallucination refers to the phenomenon where the model produces text that appears plausible yet is erroneous or not grounded in the provided input[32]. These fabricated responses or information may be untrue, inaccurate, or even contradictory to established world knowledge. Hallucinations in LLMs manifest in various forms, encompassing input-conflicting hallucination[54], content-conflicting hallucination[37], and fact-conflicting hallucination[44].

Input-conflicting hallucination arises when the generated content deviates from the user's input. For instance, when tasked with summarizing a specific document, an LLM may produce a summary that is incongruent with the document's content. Content-conflicting Hallucination occurs when the generated text contradicts previously generated information. This typically emerges in multi-turn dialogues or long-form text generation, where the model struggles to maintain contextual coherence. Fact-conflicting hallucination refers to the generation of content that does not align with known factual information. This can stem from the model's learning of erroneous knowledge during pre-training or from a failure to correctly apply that knowledge during generation. Addressing these hallucinations poses a significant challenge, necessitating a multifaceted approach that encompasses enhancing the quality of training data, refining model optimization strategies, incorporating external knowledge bases for validation, and improving the model's contextual understanding capabilities.

In Natural Language Processing (NLP), the issue of factual hallucinations emerging around LLMs has consistently garnered significant attention from researchers [32]. Initially, Tian et al.[54] first conjectured that hallucination can be caused by an encoder-decoder model generating content phrases without attending to the source. So they proposed a confidence score to ensure that the model attends to the source whenever necessary. Subsequently, Lee

et al. [37] found that existing language modeling datasets contain many near-duplicate examples and long repetitive substrings and develop two tools to deduplicate training datasets. Meanwhile, Biderman et al.[10] presented several case studies including novel results in memorization, term frequency effects on few-shot performance, and reducing gender bias. Later on, Gunasekar et al. [27] introduced phi-1, a new LLM for code, with significantly smaller size than competing models. Recently, Manakul et al.[44] proposed “SelfCheckGPT”, a simple sampling-based approach that can be used to fact-check the responses of black-box models in a zero-resource fashion, which leverages the simple idea that if an LLM has knowledge of a given concept, sampled responses are likely to be similar and contain consistent facts.

2.3 Enhancing LLMs with KGs

Enhancing LLMs with Knowledge Graphs (KGs) represents an approach aimed at augmenting the capabilities of LLMs by leveraging the structured knowledge encapsulated within KGs[26]. This integration strategy seeks to enhance the accuracy, reliability, and richness of LLMs’ performance in content generation, question answering, and complex task execution.

While LLMs exhibit formidable proficiency in processing natural language, their knowledge base primarily stems from the training data, potentially leading to knowledge gaps or inaccuracies in specific domains or factual details[8]. By incorporating KGs as external knowledge sources, LLMs can access a more comprehensive and accurate repository of factual information, thereby fostering greater accuracy in their responses and task execution. Furthermore, LLMs are prone to generating hallucinations[32]. Integrating factual knowledge from KGs into LLMs’ reasoning processes can significantly mitigate this phenomenon, bolstering their reliability. For instance, research has demonstrated that Knowledge Graph-based Refinement (KGR) frameworks can leverage factual knowledge within KGs to refine LLMs’ initial draft responses, thereby alleviating factual hallucinations during the reasoning process[26]. Moreover, KGs organize information in a graph-structured format, enabling the explicit representation of relationships and attributes between entities[33]. This structured representation facilitates LLMs’ comprehension and processing of complex semantic relationships, subsequently elevating their performance in complex reasoning tasks. In scenarios involving multi-step reasoning or the synthesis of multiple factual inputs, KGs provide LLMs with essential context and associative information, enabling them to complete the reasoning process with greater precision. Consequently, the fusion of LLMs and KGs further expands the applicability of LLMs. In domains such as question answering systems, dialogue systems, and recommender systems, LLMs augmented with KGs can generate responses or recommendations that are more accurate, diverse, and personalized. Additionally, in contexts demanding heightened accuracy and reliability, such as healthcare and legal services, KGs-enhanced LLMs offer more dependable services and support.

The research on utilizing KGs to enhance LLMs has entered a period of rapid development. As an external source of knowledge, KGs can be utilized to supplement the insufficient knowledge in LLMs[26]. In this paper, we roughly categorize KGs into standalone approaches[8, 30, 63] and LLM-related approaches[2, 4, 33]. Before

the emergence of LLMs, integrating knowledge representation into the training process using standalone KGs methods often necessitated careful design of model architectures and training methods. For example, Azzam et al.[3] presented WiseKGs, the first work that combines both client-side and server-side query optimization techniques in a truly dynamic fashion. Later on, Liu et al.[41] inspired by the progress of self-supervised learning and presented SelfKGs with efficient strategies to optimize this objective for aligning entities without label supervision.

After the emergence of LLMs, utilizing KGs to enhance LLMs has become a hot topic [26]. For example, Baek et al.[4] proposed to augment the knowledge directly in the input of LLMs with KGs. Meanwhile, Agarwal et al. [2] proposed KITLM, a novel knowledge base integration approach into language model through relevant information infusion. Recently, Jiang et al.[33] inspired by the studies on tool augmentation for LLMs and developed an Iterative Reading-then-Reasoning (IRR) framework to solve question answering tasks based on structured data, called StructGPT.

Differing from the methods mentioned above, we integrate KGs into multiple LLMs within a multi-modal environment, employing a collaborative approach among these models to tackle open-set object recognition tasks.

3 PROPOSED METHOD

To enhance the precision of open-set object recognition tasks, the core is to diminish spurious-discriminative features. Ideally, all features should be clear and explicit, capable of distinguishing all open-set classes. Our goal is to optimize algorithms to progressively approach this scenario. Motivated by the excellent work in LMC[49], our work extends upon its foundation and addresses certain limitations. Specifically, we reconstruct a collaborative framework for LLMs, addressing the illusion problem inherent in ChatGPT. Initially, we introduce KGs as an external knowledge source to rectify and augment the text descriptions generated by ChatGPT. Subsequently, we devise an image-text fusion strategy. In a multi-modal setting, leveraging relevant images generated during the LMC iteration process as references, we further optimize the text descriptions generated by ChatGPT. The workflow of our work is illustrated in Figure 2. Initially, a large number of virtual open-set classes are generated by LLMs based on a portion of existing closed-set classes. These virtual open-set classes undergo optimization through self-checking strategies employed by us. Subsequently, the closed-set classes and virtual open-set classes are jointly fed into the LLMs’ collaborative architecture for iterative optimization, resulting in the generation of textual descriptions and images. Our novel design enhances the efficiency and robustness of the iterative process. Finally, the given image is tested to determine whether it belongs to an open-set class, yielding the ultimate outcome. Next, we will discuss each key detail point in turn.

3.1 LLMs Collaboration in Cyclic

To enhance the accuracy of open-set object recognition, the core lies in reducing spurious-discriminative features. To achieve this, we leverage the powerful generative capability of LLMs to produce more discriminative features. Inspired by chain-of-thought [57],

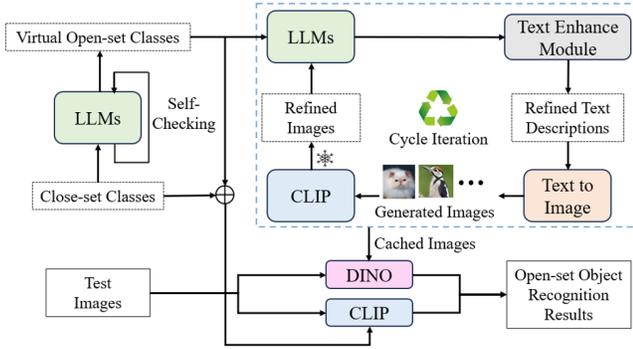


Figure 2: The overall flowchart of our proposed method. Through collaborative iteration in a cyclic manner, LLMs can continuously optimize internal knowledge, thereby enhancing the accuracy of open-set object recognition.

we set up a series of questions to facilitate LLMs in better understanding our requirements and generating the answers we desire. For a given closed-set class, we typically pose three questions to LLMs. 1) *Can you describe the features of the given class?* 2) *What other classes exhibit similar features?* 3) *Please provide distinctive features that can differentiate above classes.* Through these three questions, LLMs can provide the desired answers. For example, our dialogue with ChatGPT can be described as follows. User: *Given a list of classes [ladybug, dragonfly, goose,...], can you describe the visual features of each class in the list?* ChatGPT: *Sure. **Ladybug** is a small, round beetle that is typically red or orange with black spots. Its body is covered in a hard shiny shell. **Dragonfly** is ...* User: *What are the discriminative visual features of class **ladybug** compared with other classes in the list?* ChatGPT: *Compared with other classes in the list, ladybugs have several discriminate visual features including: **color**(ladybugs are typically red or orange with black spots); **shape**(ladybugs have a round shape);...* User: *Can you list other classes that also share these discriminative visual features?* ChatGPT: *Sure, here are some other classes that share some of the discriminative visual features of ladybugs. **Tortoise beetle**: a type of beetle that has a similar body shape and hard, shiny shell to ladybugs, and some species have similar bright colors and markings. **Ladybird spider**: a species of spider that has a similar coloration and spotted pattern to ladybugs.*

To enhance the robustness of ChatGPT against spurious discriminative features, we endeavor to create an exhaustive virtual open-set class catalog. Our goal is to empower ChatGPT with self-reflective capabilities, encouraging it to reassess whether any overlooked spurious characteristics exist. Specifically, for each closed-set category, post the initial query sequence, we augment the first question’s class list with the virtually generated open-set classes. This expansion prompts ChatGPT to revisit the three questions, effectively filtering out already-identified spurious features and prompting exploration of undetected ones. The self-checking loop iterates until ChatGPT ceases to suggest novel virtual open-set classes or a predefined maximum cycle threshold is reached. Beyond this, inspired by prior work[45] acknowledging the diversity

of real-world classes, we mimic virtual open-set instances that exhibit lower similarity to closed-set samples, thereby sharing fewer spurious discriminative traits. To accomplish this, we inquire of ChatGPT: *Considering the given class list, can you suggest classes that are distinct from these?* By integrating these simulated open-set classes into the closed-set inventory, the original discriminatory power of spurious features among closed-set classes diminishes in the expanded context.

After obtaining virtual open-set classes, we can utilize these classes to prompt LLMs to generate corresponding images, thereby complementing textual knowledge and obtaining more accurate recognition results. In particular, let the closed-set classes be denoted as Z_c and the virtual open-set classes as Z_v . For each class $z \in Z_c \cup Z_v$, we prompt a text-to-image model (i.e., DALL-E) to generate corresponding images based on their generated textual descriptions. This process can be described as follows:

$$i_z^k = T2I(d_z^k), k \in [1, K] \quad (1)$$

Where, K denotes the number of generated textual descriptions for each class. d_z^k represents the k -th textual description generated by LLMs for each class z . i_z^k represents the image generated by the text-to-image model from d_z^k .

However, images generated by LLMs may also be inaccurate. If inaccurate or erroneous images are used as references, it could lead to a decrease in the accuracy of open-set object recognition. To tackle the issue of inaccurate image representations for their intended classes, we designate these images as *less precise* and aim to automate their detection, subsequent description refinement, and image regeneration. Drawing inspiration from human learning, which incorporates feedback for enhanced understanding, we explore the potential for LLMs to refine their outputs based on peer feedback. Specifically, we introduce a cyclic cross-evaluation framework, leveraging CLIP as the *feedback generator* to inform ChatGPT of the less precise images. This enables ChatGPT to refine the descriptions associated with these images, leveraging CLIP’s insights. Therefore, we employ another LLM as an evaluator to assess the quality of generated images. If the image quality meets the criteria, it is retained. Otherwise, the image is discarded, and the next generation process begins anew. Specifically, let D denote the textual descriptions during the generation process. For each i_z^k , we utilize the multi-modal contrastive model CLIP to evaluate the quality of the generated image. The evaluation process can be represented by the following formula:

$$pass = softmax(CLIP_{vis}(i_z^k)(CLIP_{text}(D))^T) \quad (2)$$

Where, $CLIP_{vis}(\cdot)$ represents the visual operation of CLIP model, and $CLIP_{text}(\cdot)$ represents the text operation of CLIP model.

By iteratively executing the aforementioned three steps, we harness CLIP’s capability to provide targeted feedback to ChatGPT, guiding its description refinement towards our desired outcome. This refinement process results in the generation of images that more accurately represent their respective classes. The iteration concludes when all images align closely with their intended classes or a preset maximum cycle limit is reached. Notably, upon reaching this limit, any remaining images deemed inaccurate are discarded.

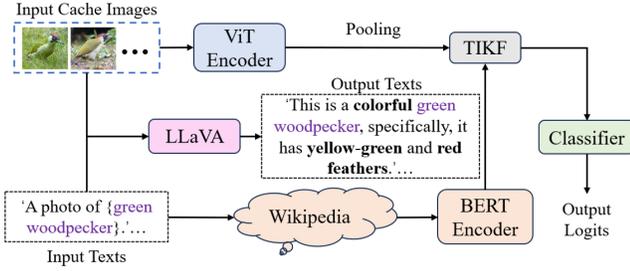


Figure 3: Diagram illustrating our proposed text enhance module. The input text is sourced from generative language models, while the input images are derived from images generated during the iteration process. TIKF represents our proposed Text-Image Knowledge Fusion strategy. The output consists of two parts: one part comprises textual descriptions used for image generation, and the other part consists of probability values used for subsequent evaluation.

3.2 Textual Enhance Module

While the collaboration of LLMs has shown astonishing effectiveness, LLMs also possess their own limitations. For instance, the common issue of hallucinations in generative dialogue models such as ChatGPT. The occurrence of hallucinations during the iteration process constitutes a critical challenge, as it can significantly impede the results of open-set target recognition. For generative language models operating solely in a unimodal fashion, their capability is inherently constrained. Some responses may appear logically correct at first glance, yet upon further examination, it becomes apparent contradicts common knowledge. To address this issue, we draw inspiration from the operational mode of collaborative LLMs and propose a Text-Image multi-modal Knowledge Fusion (TIKF) strategy. By using images as references, we rectify erroneous responses from generative language models. Furthermore, as generative language models are internally pre-trained, their knowledge base is fixed. Moreover, human language exhibits many instances of polysemy, which LLMs struggle to disambiguate. To mitigate this issue, we introduce a KGs as an external knowledge source, compensating for this limitation of LLMs. The specific process is illustrated in Figure 3.

Our method takes textual description D and image i_z^k as input. The image i_z^k is first encoded by a Vision Transformer (ViT) to obtain the image vector iv_z^k . Due to the presence of ambiguity and incomplete content in D , we introduce a KGs to rectify and complement it.

The KGs is a structured semantic knowledge base that depicts concepts, entities, and their relationships in the objective world through a graphical representation. It transforms the vast information on the internet into a form that more closely aligns with human cognition, enabling enhanced capabilities for organizing, managing, and comprehending this deluge of data. While Wikipedia can be considered as an illustrative manifestation of a KG, it transcends mere visualization by leveraging semantic technologies to comprehend and represent knowledge, thereby supporting semantic search

and reasoning[53]. Moreover, its structured storage and representation of knowledge facilitate efficient processing and analysis by computers. With the advent of web technologies and artificial intelligence, the evolution of Wikipedia has accelerated significantly. The concepts and technologies underlying it have continually progressed from semantic networks, ontologies, linked data, to the current KG paradigm. Typically, Wikipedia comprises entities, attributes, and relations. Entities correspond to semantic instances, such as "Yao Ming" or "China," constituting the fundamental units of the graph. Attributes describe the characteristics of an entity class, for instance, "height" being an attribute of Yao Ming with a value of "229 centimeters." Relations, on the other hand, articulate the connections between semantic instances, linking entities like "Yao Ming" and "China" through the relation "nationality."

In this paper, we utilize Wikipedia as an external knowledge source. The processed textual description is encoded by BERT[12] into a text vector D_v . The above process can be represented by the following formula:

$$iv_z^k = ViT(i_z^k), k \in [1, K] \quad (3)$$

$$D_v = BERT(Wiki(D)) \quad (4)$$

Where $ViT(\cdot)$ denotes the encoding operation using the ViT model, $Wiki(\cdot)$ represents the online matching and correction using Wikipedia, and $BERT(\cdot)$ indicates the encoding operation using the BERT model.

Afterward, we propose a strategy to integrate image and text knowledge. By appropriately integrating iv_z^k and D_v , and jointly inputting them into the classifier, more accurate classification results can be obtained. Since the dimension of iv_z^k is much higher than that of D_v , we performed pooling and dimension reduction on iv_z^k before fusion to better match D_v . Additionally, inspired by the advanced multi-modal LLaVA[39], we utilized it to simultaneously accept image and text inputs. This approach allows for obtaining more detailed and accurate textual descriptions D_l from another perspective, which are directly used for generating high-quality images. The computational process is illustrated as follows:

$$D_l = LLaVA(D, i_z^k), k \in [1, K] \quad (5)$$

Here, $LLaVA(\cdot, \cdot)$ represents the utilization of the LLaVA model for processing input data.

3.3 Text-Image Knowledge Fusion

To better leverage the rich semantic knowledge contained within images for aiding in text correction and completion, we propose TIKF, where image vectors and text vectors are appropriately integrated to jointly determine the category logits of the object, as illustrated in Figure 4.

We continue the discussion from the previous subsection. The input for the TIKF strategy consists of image embeddings iv_z^k and Wiki entity embeddings D_v . Firstly, we perform a linear operation on the input iv_z^k to reweight the input image vectors and text vectors, enhancing the integration of their information. Subsequently, we perform averaging on D_v to mitigate differences between Wiki entity lengths, facilitating better fusion with the image vectors. This process can be represented by the following formula:

$$ivl_z^k = LN(iv_z^k), k \in [1, K] \quad (6)$$

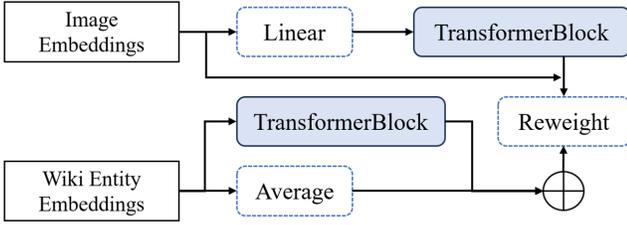


Figure 4: Figure illustrating the TIKF strategy. We input image embeddings and Wiki entity embeddings, which undergo transformer blocks and are subjected to operations such as linear transformation, averaging, and reweighting, to accomplish text-image knowledge fusion.

$$\bar{D}_v = \text{AVG}(D_v) \quad (7)$$

Where iv_z^k represents the data obtained after linear calculation of iv_z^k , and \bar{D}_v denotes the result after averaging D_v . $LN(\cdot)$ denotes the linear operation, and $\text{AVG}(\cdot)$ represents the averaging operation.

To retain details and multiscale features, we opt to linearize and average only a portion of iv_z^k and D_v , leaving the other part untreated. It is noteworthy that the data after linearization needs further processing with TransformerBlock, whereas only the data untreated by averaging requires subsequent TransformerBlock processing. Finally, we perform element-wise addition between D_v and \bar{D}_v , and reweight them together with iv_z^k and iv_z^k to obtain the final result D_f . This process can be described as follows:

$$D_f = \text{Rwt}(iv_z^k, iv_z^k, D_v \oplus \bar{D}_v), k \in [1, K] \quad (8)$$

Where $\text{Rwt}(\cdot)$ represents the operation of reweighting, and \oplus denotes element-wise addition.

3.4 Inference

After collaborative efforts from LLMs, we obtained a plethora of high-quality textual descriptions and images. We have augmented the closed-set class list with virtual open-set classes and produced a diverse array of images for each class in this expanded list. This section outlines how our framework leverages this expanded dataset and generated images to diminish the discriminative power of spurious features during inference. Notably, our framework operates without prior training, enabling seamless integration into real-world applications.

Inspired by the image-text contrastive model CLIP[1] and the image-image contrastive model DINO[61], we employ these two models jointly to accomplish the open-set object recognition task. To ensure the efficiency of LLMs during inference, we pre-store the image and text features required by CLIP and DINO before conducting inference. After completing these steps, we can proceed with the inference task, which can be divided into three parts. The first part involves obtaining an assessment probability using the CLIP model. The second part entails obtaining a probability value using the DINO model. The third part involves weighting and summing these two probability values to obtain the final open-set object recognition result. We will now elaborate on each of these three parts.

First, given a test image I_t , we input it into the image encoder of CLIP. Then, we input the pre-stored text classes Z_c and Z_{vo} , which are summed element-wise, into the text encoder of CLIP. This process yields the prediction results of the CLIP model P_c :

$$P_c = \text{softmax}(\text{CLIP}_{vis}(I_t)(\text{CLIP}_{text}((Z_c \oplus Z_{vo})))^T) \quad (9)$$

Similarly, we utilize the two image encoders of DINO to respectively receive the test image I_t and the pre-stored generated image i_z^k . After averaging and softmaxing, we obtain the prediction probability values P_d of DINO:

$$P_d = \text{softmax}(\text{AVG}(\text{DINO}_{vis}(I_t)(\text{DINO}_{vis}(i_z^k))^T)), k \in [1, K] \quad (10)$$

Where $\text{DINO}_{vis}(\cdot)$ represents the visual operation of the DINO model.

Finally, we weight and sum the two probabilities P_c and P_d , then calculate the maximum probability to obtain the final open-set object recognition result P_{all} :

$$P_{all} = \max_{z \in Z_c} (mP_c + (1 - m)P_d) \quad (11)$$

4 EXPERIMENTS

4.1 Dataset and Metrics

We evaluate our method on three datasets: CIFAR10[36], CIFAR10+[36] and TinyImageNet [19].

The CIFAR10 dataset is a classic dataset used for image classification tasks. It consists of color images categorized into 10 classes, each containing 6000 images sized 32×32 pixels. These classes include airplanes, automobiles, birds, cats, deer, dogs, frogs, horses, ships, and trucks. The images in CIFAR10 are divided into 50,000 training images and 10,000 test images. This dataset is widely utilized for evaluating the performance of image classification algorithms, particularly in research within the field of computer vision. In this paper, following LMC [49], we randomly select 6 closed-set classes and 4 open-set classes for evaluation. Similarly, following LMC, we expand the CIFAR10 dataset by randomly selecting additional 10 classes from the full CIFAR set, incorporating them into CIFAR10 to form CIFAR10+, in order to better evaluate our method.

TinyImageNet is a downscaled version of the ImageNet dataset designed for image related tasks. It serves as a subset of ImageNet, comprising approximately 100,000 training images, about 10,000 validation images, and roughly 10,000 test images. Each image has a size of 64×64 pixels, and the dataset consists of 200 categories. For each category, there are 500 training images, 50 validation images, and 50 test images. TinyImageNet aims to provide a relatively small yet challenging dataset suitable for rapid prototyping and model validation. In this paper, following [49], we randomly select 20 closed-set classes and 180 open-set classes for evaluation.

In this paper, we follow the approach of [49], employing AUROC and OSCR[20] as the evaluation metrics for our method.

The AUROC, an acronym for Area Under the Receiver Operating Characteristic curve, represents the area beneath the Receiver Operating Characteristic (ROC) curve, serving as a pivotal metric for evaluating classifier performance. This metric encapsulates the classifier's efficacy by quantifying the area between the ROC curve and the coordinate axes. Fundamentally, the AUROC signifies the expectation that a randomly drawn positive instance (true positive)

will rank higher than a randomly drawn negative instance (true negative). As a value ranging from 0 to 1, an AUROC score approaching 1 indicates a classifier’s exceptional ability to distinguish between positive and negative samples. In the realm of machine learning and statistical classification, AUROC is widely employed for assessing model performance, particularly adept at handling imbalanced datasets due to its invariance to the ratio of positive to negative samples.

In accordance with the definition provided in [20], we utilize OSCR (Open-Set Classification Rate curve, assuming this as a hypothetical metric for the sake of the context) as an evaluation metric to test the performance of our method.

4.2 Implementation Details

In this paper, we collaborate multiple LLM in a plug-and-play manner without the necessity for additional training processes. During the feature optimization phase, we set the maximum self-check iterations for LLMs at 3. In the process of generating textual descriptions, we rank the optimized textual descriptions by accuracy from high to low. For each category, we select the top 10 descriptions. To prevent the algorithm from falling into an infinite loop, we set the maximum number of iterations for feature optimization to 3 times. For the CLIP model, we employ ViT-B/32 as the image encoder and a general transformer structure as the text encoder. For the DINO model, we use ViT-B/14 as the image encoder. Additionally, we set m to 0.6 for the calculation of P_{all} .

The computer hardware configuration we utilized includes a 13th Gen Intel(R) Core(TM) i5-13490F 2.50 GHz CPU, 32.0 GB RAM, and one NVIDIA RTX 3090ti GPU. Our code execution environment comprises Python 3.9, PyTorch 2.0.0, and CUDA 11.7.

4.3 Comparison with Other Methods

To demonstrate the efficacy of our method, we specifically chose to compare it with the leading algorithms published in core articles/papers in recent years. The articles/papers include ICCV, ECCV, CVPR, TPAMI, AAAI, ICLR, and NeurIPS. Moreover, all comparison methods except [49] require additional training. Our proposed method do not require any extra training processes. This can save a significant amount of computational resources, and our code can easily be run on general computing devices. As evidenced in Table 1 and Table 2, our method outperforms all the compared methods, achieving the best performance.

Compared to previous methods, our approach has achieved a new state-of-the-art performance. Furthermore, in terms of error fluctuations, our method exhibits remarkable stability, indicating that our proposed open-set object recognition method is highly robust and largely immune to sample variations. We attribute this primarily to the capability of LLMs to compensate for missing samples and generate superior samples to replace inferior ones, thereby ensuring the quality of samples and consequently, the performance of the algorithm. In comparison to [49], our approach surpasses it in terms of average performance and significantly outperforms it in error fluctuations. We believe this is due to our utilization of knowledge graphs to mitigate the hallucination issues inherent in LLM, a method that holds significant importance for enhancing the output quality of LLMs.

It is noteworthy that our method excels in error reduction. Compared to other methods, our approach has successfully reduced the error by a factor of 10 to 100. We attribute this phenomenon to the error correction and completion capabilities enabled by multi modalities and external Knowledge Graphs, which effectively rectify the erroneous outputs of LLMs, resulting in stable and correct outputs.

4.4 Ablation Studies

In this subsection, we progressively analyze the role of each module. Following the methodology in [49], we use AUROC as the evaluation metric. Specifically, the dataset employed is TinyImageNet.

First, we analyzed the respective roles of text classes and images in open-set object detection during the inference stage. The text classes originate from initial classes and virtual open-set classes generated after self-checking strategies by LLMs. One part of the images is test images, while the other part comes from images generated during the iterative cycling process. Our test code environment excludes these two parts separately and yields varying results, as shown in Table 3. Among them, ‘w/o Text Classes’ indicates that in our ablation study, we did not include text classes for inference, and only images were involved throughout the inference step. Under this scenario, the results of open-set object recognition lacked references from text classes, thus leading to a decrease in recognition accuracy. Moreover, ‘w/o images’ signifies that we did not incorporate relevant images for inference in our ablation experiment, and only text classes participated in the entire inference process. The absence of relevant images as references also limits the performance of the overall recognition algorithm. To make the results of the ablation study clearer, we specifically included the method from [49] for comparison, which further demonstrates that our method has made improvements and enhancements to it, indeed playing a positive role.

Next, we explore the roles played by different modules in the overall process. In this discussion, we will address the overall impact of the four components: self-checking, cycling generation, text-enhancing, and the Knowledge Graphs (KGs) on the results.

The results of the ablation studies among the various modules are shown in Table 4. We employed a stepwise incremental strategy by progressively adding the test modules. Overall, as the number of modules increases, the results of open-set object recognition become more precise, indicating that each module contributes positively to the efficacy of the entire method. Specifically, we found that adding the Knowledge Graphs (KGs) module alone results in a better improvement than adding the text-enhancing module alone. We analyze this situation might due to the knowledge relied upon by the text-enhancing module is primarily derived from within LLMs themselves, which inherently has certain limitations. In contrast, the Knowledge Graphs is different, which benefits from the external Internet and can be updated in real time. Hence, it can supplement latest knowledge that LLMs could not reach. This also proves that the Knowledge Graphs becomes a powerful tool to compensate for the knowledge base of LLMs.

Table 1: A comparison of our method with the most optimal algorithms in recent years in terms of the AUROC metric, wherein the best indicators are highlighted in bold for emphasis.

Methods	Source	CIFAR10	CIFAR10+	TinyImageNet
Neal et al.[46]	ECCV 2018	69.9±3.8	83.8±-	58.6±-
Oza et al.[47]	CVPR 2019	89.5±-	95.5±-	74.8±-
Chen et al.[15]	ECCV 2020	90.1±-	97.6±-	80.9±-
Zhang et al.[60]	ECCV 2020	95.0±-	96.2±-	79.3±-
Guo et al.[28]	ICCV 2021	83.5±2.3	88.8±1.9	71.5±1.8
Chen et al.[14]	TPAMI 2022	90.1±0.5	96.5±0.6	76.2±0.5
Chen et al.2[14]	TPAMI 2022	91.0±0.7	97.1±0.3	78.2±1.3
Lu et al.[43]	AAAI 2022	95.1±-	97.8±-	83.1±-
Esmailpour et al.[22]	AAAI 2022	93.0±1.7	97.8±0.6	84.6±1.0
Vaze et al.[55]	ICLR 2022	93.6±-	97.9±-	83.0±-
Moon et al.[45]	ECCV 2022	85.0±2.2	92.0±1.1	73.1±1.5
Cho et al.[18]	ECCV 2022	94.8±-	96.1±-	78.5±-
Liu et al.[42]	TPAMI 2023	85.7±1.3	89.1±1.4	76.4±1.7
Liu et al.2[42]	TPAMI 2023	88.5±1.3	91.8±0.8	74.6±0.8
Huang et al.[29]	TPAMI 2023	91.3±-	96.3±-	82.3±-
Huang et al.2[29]	TPAMI 2023	91.5±-	96.0±-	81.9±-
Qu et al.[49]	NeurIPS 2023	96.6±0.3	98.9±0.7	86.7±1.4
Ours		98.4±0.02	99.5±0.002	89.2±0.03

Table 2: A comparison of our method with the best algorithms from recent years based on the OSCR metric, with the highest indicators prominently displayed in bold.

Methods	Source	CIFAR10	CIFAR10+	TinyImageNet
Yang et al.[58]	CVPR 2018	84.3±1.7	91.0±1.7	59.3±5.3
Chen et al.[15]	ECCV 2020	85.2±1.4	91.8±1.2	53.2±4.6
Chen et al.[14]	TPAMI 2022	86.6±1.4	93.5±0.8	62.3±3.3
Chen et al.2[14]	TPAMI 2022	87.9±1.5	94.7±0.7	65.9±3.8
Liu et al.[42]	TPAMI 2023	84.8±1.4	92.5±1.0	64.3±3.2
Liu et al.2[42]	TPAMI 2023	86.9±1.5	93.2±0.3	59.2±2.1
Qu et al.[49]	NeurIPS 2023	93.6±1.5	96.8±0.7	80.6±3.4
Ours		95.9±0.02	96.9±0.008	82.9±0.05

Table 3: Ablation Study of Alignment Strategies.

Strategy	AUROC
Images	84.4
Text Classes	82.1
Qu et al.[49]	86.7
Ours	89.2

Table 4: Ablation Study of Different Modules.

Module	AUROC
self-checking	85.6
cycling	85.9
self-checking + cycling	86.7
self-checking + cycling + text-enhancing	87.6
self-checking + cycling + KGs	88.5
Ours	89.2

4.5 Visualization

In this subsection, we provide specific examples regarding text-enhancement and image optimization, more vividly illustrating the effectiveness and superiority of our method. Details shown in Figure 5.

From the visual examples, it can be clearly seen that the enhanced text descriptions through multi-modal and Knowledge Graphs techniques are more specific and contain more detailed information, such as the appearance and location of objects in the images. Additionally, the enhanced text descriptions include some inferential contents, which greatly aid in the deep understanding of images and complex image generation tasks. For the optimized images, we observe that our proposed method can generate images that match the text descriptions to a greater extent. This underscores the superiority of our method.

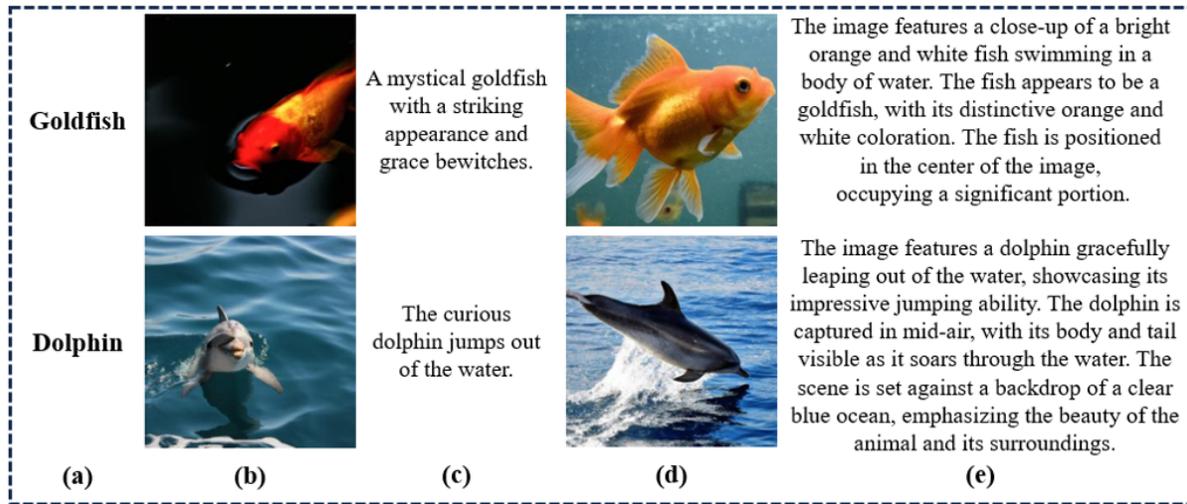


Figure 5: Visual schematics for text enhancement and image optimization. Herein, (a) - (e) correspond to the class name, the image before optimization, the text description before error correction and completion, the image after optimization, and the text description after error correction and completion, respectively.

5 CONCLUSION

In this paper, we propose an optimized collaboration framework of LLMs for the task of open-set object recognition, achieving remarkable results. Addressing the hallucination issue prevalent in textual LLMs, we introduce a solution that leverages multi-modalities and Knowledge Graphs. More specifically, we devised an integrated method that combines textual and image knowledge. In the future, as LLMs evolve, our method is expected to demonstrate even better performance.

ACKNOWLEDGMENTS

This work was supported by the High Performance Computing Platform of Hohai University.

REFERENCES

- [1] Rabab Abdelfattah, Qing Guo, Xiaoguang Li, Xiaofeng Wang, and Song Wang. 2023. CDUL: CLIP-Driven Unsupervised Learning for Multi-Label Image Classification. In *Proceedings of IEEE International Conference on Computer Vision*. 1348–1357.
- [2] Ankush Agarwal, Saksham Gawade, Amar Prakash Azad, and Pushpak Bhattacharyya. 2023. KITLM: Domain-Specific Knowledge InTegration into Language Models for Question Answering. *CoRR abs/2308.03638* (2023).
- [3] Amr Azzam, Christian Aebelo, Gabriela Montoya, Ilkcan Keles, Axel Polleres, and Katja Hose. 2021. WiseKG: Balanced Access to Web Knowledge Graphs. In *Proceedings of International Conference on World Wide Web*. 1422–1434.
- [4] Jinheon Baek, Alham Fikri Aji, and Amir Saffari. 2023. Knowledge-Augmented Language Model Prompting for Zero-Shot Knowledge Graph Question Answering. *CoRR abs/2306.04136* (2023).
- [5] Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwel Ji, Tiezheng Yu, Willy Chung, Quyet V. Do, Yan Xu, and Pascale Fung. 2023. A Multitask, Multilingual, Multimodal Evaluation of ChatGPT on Reasoning, Hallucination, and Interactivity. In *Proceedings of International Joint Conference on Natural Language Processing*. 675–718.
- [6] Jose Barambones, Cristian Moral, Angélica de Antonio, Ricardo Imbert, Loïc Martínez Normand, and Elena Villalba Mora. 2024. ChatGPT for Learning HCI Techniques: A Case Study on Interviews for Personas. *IEEE Trans. Learn. Technol.* 17 (2024), 1486–1501.
- [7] Onur Barut, Yan Luo, Peilong Li, and Tong Zhang. 2023. R1DIT: Privacy-Preserving Malware Traffic Classification With Attention-Based Neural Networks. *IEEE Trans. Netw. Serv. Manag.* 20, 2 (2023), 2071–2085.
- [8] Anson Bastos, Abhishek Nadgeri, Kuldeep Singh, Isaiah Onando Mulang, Saeedeh Shekarpour, Johannes Hoffart, and Manohar Kaul. 2021. RECON: Relation Extraction using Knowledge Graph Context in a Graph Neural Network. In *Proceedings of International Conference on World Wide Web*. 1673–1685.
- [9] Abhijit Bendale and Terrance E. Boult. 2016. Towards Open Set Deep Networks. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. 1563–1572.
- [10] Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O’Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar van der Wal. 2023. Pythia: A Suite for Analyzing Large Language Models Across Training and Scaling. In *Proceedings of International Conference on Machine Learning*. 2397–2430.
- [11] Boxi Cao, Hongyu Lin, Xianpei Han, Le Sun, Lingyong Yan, Meng Liao, Tong Xue, and Jin Xu. 2021. Knowledgeable or Educated Guess? Revisiting Language Models as Knowledge Bases. In *Proceedings of Association for Computational Linguistics and International Joint Conference on Natural Language Processing*. 1860–1874.
- [12] Haw-Shiuan Chang, Rwei-Yao Sun, Kathryn Ricci, and Andrew McCallum. 2023. Multi-CLS BERT: An Efficient Alternative to Traditional Ensembling. In *Proceedings of Conference on Association for Computational Linguistics*. 821–854.
- [13] Anthony Chen, Panupong Pasupat, Sameer Singh, Hongrae Lee, and Kelvin Guu. 2023. PURR: Efficiently Editing Language Model Hallucinations by Denoising Language Model Corruptions. *CoRR abs/2305.14908* (2023).
- [14] Guangyao Chen, Peixi Peng, Xiangqian Wang, and Yonghong Tian. 2022. Adversarial Reciprocal Points Learning for Open Set Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* 44, 11 (2022), 8065–8081.
- [15] Guangyao Chen, Limeng Qiao, Yemin Shi, Peixi Peng, Jia Li, Tiejun Huang, Shiliang Pu, and Yonghong Tian. 2020. Learning Open Set Network with Discriminative Reciprocal Points. In *Proceedings of European Conference on Computer Vision*. 507–522.
- [16] Ruirui Chen, Chengwei Qin, Weifeng Jiang, and Dongkyu Choi. 2024. Is a Large Language Model a Good Annotator for Event Extraction?. In *Proceedings of AAAI Conference on Artificial Intelligence*. 17772–17780.
- [17] Yuhan Chen, Nuwa Xi, Yanrui Du, Haochun Wang, Jianyu Chen, Sendong Zhao, and Bing Qin. 2024. From Artificially Real to Real: Leveraging Pseudo Data from Large Language Models for Low-Resource Molecule Discovery. In *Proceedings of AAAI Conference on Artificial Intelligence*. 21958–21966.
- [18] Wonwoo Cho and Jaegul Choo. 2022. Towards Accurate Open-Set Recognition via Background-Class Regularization. In *Proceedings of European Conference on Computer Vision*, Vol. 13685. 658–674.
- [19] Jia Deng. 2009. A large-scale hierarchical image database. *Proceedings of IEEE Computer Vision and Pattern Recognition* (2009).
- [20] Akshay Raj Dhamija, Manuel Günther, and Terrance E. Boult. 2018. Reducing Network Agnostophobia. In *Proceedings of Conference on Neural Information Processing Systems*. 9175–9186.

- [21] Xuefeng Du, Xin Wang, Gabriel Gozum, and Yixuan Li. 2022. Unknown-Aware Object Detection: Learning What You Don't Know from Videos in the Wild. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. 13668–13678.
- [22] Sepideh Esmailpour, Bing Liu, Eric Robertson, and Lei Shu. 2022. Zero-Shot Out-of-Distribution Detection Based on the Pre-trained Model CLIP. In *Proceedings of AAAI Conference on Artificial Intelligence*. 6568–6576.
- [23] Yimin Fu, Zhunga Liu, and Zuowei Zhang. 2023. Progressive Learning Vision Transformer for Open Set Recognition of Fine-Grained Objects in Remote Sensing Images. *IEEE Trans. Geosci. Remote. Sens.* 61 (2023), 1–13.
- [24] Pengen Gao, Shengkai Zhang, Wei Wang, and Chris Xiaoxuan Lu. 2024. Robust Metric Localization in Autonomous Driving via Doppler Compensation With Single-Chip Radar. *IEEE Trans. Intell. Transp. Syst.* 25, 1 (2024), 491–502.
- [25] Zongyuan Ge, Sergey Demyanov, and Rahil Garnavi. 2017. Generative OpenMax for Multi-Class Open Set Classification. In *Proceedings of British Machine Vision Conference*.
- [26] Xinyan Guan, Yanjiang Liu, Hongyu Lin, Yaojie Lu, Ben He, Xianpei Han, and Le Sun. 2024. Mitigating Large Language Model Hallucinations via Autonomous Knowledge Graph-Based Retrofitting. In *Proceedings of AAAI Conference on Artificial Intelligence*. 18126–18134.
- [27] Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, Adil Salim, Shital Shah, Harkirat Singh Behl, Xin Wang, Sébastien Bubeck, Ronen Eldan, Adam Tauman Kalai, Yin Tat Lee, and Yuanzhi Li. 2023. Textbooks Are All You Need. *CoRR abs/2306.11644* (2023).
- [28] Yunrui Guo, Guglielmo Camporese, Wenjing Yang, Alessandro Sperduti, and Lamberto Ballan. 2021. Conditional Variational Capsule Network for Open Set Recognition. In *Proceedings of IEEE International Conference on Computer Vision*. 103–111.
- [29] Hongzhi Huang, Yu Wang, Qinghua Hu, and Ming-Ming Cheng. 2023. Class-Specific Semantic Reconstruction for Open Set Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* 45, 4 (2023), 4214–4228.
- [30] Jiacheng Huang, Yao Zhao, Wei Hu, Zhen Ning, Qijin Chen, Xiaoxia Qiu, Chengfu Huo, and Weijun Ren. 2022. Trustworthy Knowledge Graph Completion Based on Multi-sourced Noisy Data. In *Proceedings of International Conference on World Wide Web*. 956–965.
- [31] Liming Huang and Aojun Gong. 2024. Surface Defect Detection for No-Service Rails With Skeleton-Aware Accurate and Fast Network. *IEEE Trans. Ind. Informatics* 20, 3 (2024), 4571–4581.
- [32] Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Yejin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of Hallucination in Natural Language Generation. *ACM Comput. Surv.* 55, 12 (2023), 248:1–248:38.
- [33] Jinhao Jiang, Kun Zhou, Zican Dong, Keming Ye, Xin Zhao, and Ji-Rong Wen. 2023. StructGPT: A General Framework for Large Language Model to Reason over Structured Data. In *Proceedings of Conference on Empirical Methods in Natural Language Processing*. 9237–9251.
- [34] Nazmul Karim, Niluthpol Chowdhury Mithun, Abhinav Rajvanshi, Han-Pang Chiu, Supun Samarasekera, and Nazanin Rahnavard. 2023. C-SFDA: A Curriculum Learning Aided Self-Training Framework for Efficient Source Free Domain Adaptation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. 24120–24131.
- [35] Shu Kong and Deva Ramanan. 2021. OpenGAN: Open-Set Recognition via Open Data Generation. In *Proceedings of IEEE International Conference on Computer Vision*. 793–802.
- [36] Alex Krizhevsky, Geoffrey Hinton, et al. 2009. Learning multiple layers of features from tiny images. (2009).
- [37] Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. 2022. Deduplicating Training Data Makes Language Models Better. In *Proceedings of Meeting of the Association for Computational Linguistics*. 8424–8445.
- [38] Hang Li, Jindong Gu, Rajat Koner, Sahand Sharifzadeh, and Volker Tresp. 2023. Do DALL-E and Flamingo Understand Each Other?. In *Proceedings of IEEE International Conference on Computer Vision*. 1999–2010.
- [39] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual Instruction Tuning. In *Proceedings of Conference on Neural Information Processing Systems*.
- [40] Hammeng Liu, Ruoxi Ning, Zhiyang Teng, Jian Liu, Qiji Zhou, and Yue Zhang. 2023. Evaluating the Logical Reasoning Ability of ChatGPT and GPT-4. *CoRR abs/2304.03439* (2023).
- [41] Xiao Liu, Haoyun Hong, Xinghao Wang, Zeyi Chen, Evgeny Kharlamov, Yuxiao Dong, and Jie Tang. 2022. SelfKG: Self-Supervised Entity Alignment in Knowledge Graphs. In *Proceedings of International Conference on World Wide Web*. 860–870.
- [42] Zhun-ga Liu, Yimin Fu, Quan Pan, and Zuowei Zhang. 2023. Orientational Distribution Learning With Hierarchical Spatial Attention for Open Set Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* 45, 7 (2023), 8757–8772.
- [43] Jing Lu, Yunlu Xu, Hao Li, Zhanzhan Cheng, and Yi Niu. 2022. PMAL: Open Set Recognition via Robust Prototype Mining. In *Proceedings of AAAI Conference on Artificial Intelligence*. 1872–1880.
- [44] Potsawee Manakul, Adian Liusie, and Mark J. F. Gales. 2023. SelfCheckGPT: Zero-Resource Black-Box Hallucination Detection for Generative Large Language Models. In *Proceedings of Conference on Empirical Methods in Natural Language Processing*. 9004–9017.
- [45] WonJun Moon, Jun Ho Park, Hyun Seok Seong, Cheol Ho Cho, and Jae Pil Heo. 2022. Difficulty-Aware Simulator for Open Set Recognition. In *Proceedings of European Conference on Computer Vision*. Vol. 13685. 365–381.
- [46] Lawrence Neal, Matthew L. Olson, Xiaoli Z. Fern, Weng Keen Wong, and Fuxin Li. 2018. Open Set Learning with Counterfactual Images. In *Proceedings of European Conference on Computer Vision*. Vol. 11210. 620–635.
- [47] Poojan Oza and Vishal M. Patel. 2019. C2AE: Class Conditioned Auto-Encoder for Open-Set Recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. 2307–2316.
- [48] Pramuditha Perera, Vlad I. Morariu, Rajiv Jain, Varun Manjunatha, Curtis Wigington, Vicente Ordonez, and Vishal M. Patel. 2020. Generative-Discriminative Feature Representations for Open-Set Recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. 11811–11820.
- [49] Haoxuan Qu, Xiaofei Hui, Yujun Cai, and Jun Liu. 2023. LMC: Large Model Collaboration with Cross-assessment for Training-Free Open-Set Object Recognition. In *Proceedings of Conference on Neural Information Processing Systems*.
- [50] Raha Razavi, Gerlind Plonka, and Hossein Rabbani. 2024. X-Let's Atom Combinations for Modeling and Denoising of OCT Images by Modified Morphological Component Analysis. *IEEE Trans. Medical Imaging* 43, 2 (2024), 760–770.
- [51] Lei Shu, Liangchen Luo, Jayakumar Hoskere, Yun Zhu, Yinxiao Liu, Simon Tong, Jindong Chen, and Lei Meng. 2024. RewriteLM: An Instruction-Tuned Large Language Model for Text Rewriting. In *Proceedings of AAAI Conference on Artificial Intelligence*. 18970–18980.
- [52] Xin Sun, Zhenning Yang, Chi Zhang, Keck Voon Ling, and Guohao Peng. 2020. Conditional Gaussian Distribution Learning for Open Set Recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. 13477–13486.
- [53] Gokul Thota and Vasudeva Varma. 2024. A Category-agnostic Graph Attention-based Approach for Determining Notability of Articles for Wikipedia. In *Proceedings of the ACM Web Conference*. 911–914.
- [54] Ran Tian, Shashi Narayan, Thibault Sellam, and Ankur P. Parikh. 2019. Sticking to the Facts: Confident Decoding for Faithful Data-to-Text Generation. *CoRR abs/1910.08684* (2019).
- [55] Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. 2022. Open-Set Recognition: A Good Closed-Set Classifier is All You Need. In *Proceedings of International Conference on Learning Representations*.
- [56] Cunxiang Wang, Pai Liu, and Yue Zhang. 2021. Can Generative Pre-trained Language Models Serve As Knowledge Bases for Closed-book QA?. In *Proceedings of Association for Computational Linguistics and International Joint Conference on Natural Language Processing*. 3241–3251.
- [57] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In *Proceedings of Conference on Neural Information Processing Systems*.
- [58] Hong-Ming Yang, Xu-Yao Zhang, Fei Yin, and Cheng-Lin Liu. 2018. Robust Classification With Convolutional Prototype Learning. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. 3474–3482.
- [59] Hailemariam Mehari Yohannes and Toshiyuki Amagasa. 2022. Named-entity recognition for a low-resource language using pre-trained language model. In *Proceedings of Symposium on Applied Computing*. 837–844.
- [60] Hongjie Zhang, Ang Li, Jie Guo, and Yanwen Guo. 2020. Hybrid Models for Open Set Recognition. In *Proceedings of European Conference on Computer Vision*. 102–117.
- [61] Junyi Zhang, Charles Herrmann, Junhwa Hur, Luisa Polania Cabrera, Varun Jampani, Deqing Sun, and Ming-Hsuan Yang. 2023. A Tale of Two Features: Stable Diffusion Complements DINO for Zero-Shot Semantic Correspondence. In *Proceedings of Conference on Neural Information Processing Systems*.
- [62] Renrui Zhang, Xiangfei Hu, Bohao Li, Siyuan Huang, Hanqiu Deng, Yu Qiao, Peng Gao, and Hongsheng Li. 2023. Prompt, Generate, Then Cache: Cascade of Foundation Models Makes Strong Few-Shot Learners. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. 15211–15222.
- [63] Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. ERNIE: Enhanced Language Representation with Informative Entities. In *Proceedings of Conference on Association for Computational Linguistics*. 1441–1451.
- [64] Qihao Zhao, Yalun Dai, Hao Li, Wei Hu, Fan Zhang, and Jun Liu. 2024. LTGC: Long-tail Recognition via Leveraging LLMs-driven Generated Content. *CoRR abs/2403.05854* (2024).