AIMA at SemEval-2025 Task 1: Bridging Text and Image for Idiomatic Knowledge Extraction via Mixture of Experts

Arash Rasouli[1,*], Erfan Sadraiye[1,*], Omid Ghahroodi[2,+], Hamid R. Rabiee[1,+], Ehsaneddin Asgari[2,+]

[1]Sharif University of Technology
[2]Qatar Computing Research Institute, Doha, Qatar

* These authors contributed equally to this work.

+ Joint Corresponding Authorship.

Abstract

Idioms are integral components of language, playing a crucial role in understanding and processing linguistic expressions. Although extensive research has been conducted on the comprehension of idioms in the text domain, their interpretation in multi-modal spaces remains largely unexplored. In this work, we propose a multi-expert framework to investigate the transfer of idiomatic knowledge from the language to the vision modality. Through a series of experiments, we demonstrate that leveraging text-based representations of idioms can significantly enhance understanding of the visual space, bridging the gap between linguistic and visual semantics.

1 Introduction

Idioms, such as "kick the bucket" or "spill the beans," are a common subset of multi-word expressions (MWEs) that play a crucial role in natural language understanding. MWEs are sequences of words that exhibit idiosyncratic properties, meaning their overall meaning cannot always be inferred from the meanings of their components (Sag et al., 2002). Studying idioms and MWEs is essential in natural language processing (NLP), machine translation, and sentiment analysis. Traditional NLP models often struggle with idioms since their literal interpretation differs from their intended meaning.

In recent years, the rapid advancement of deep learning and large language models has sparked significant interest in the study of idioms. Most research in this area has primarily focused on machine translation (Dankers et al., 2022; Baziotis et al., 2023; Donthi et al., 2025) and semantic analysis (Tahayna et al., 2022), yielding promising results. While significant progress has been made in understanding idioms within the textual domain, their representation in a multi-modal context remains largely unexplored.

In Task 1 of SemEval-2025 (Pickard et al., 2025), we must receive some images, a sentence, and a phrase used in it as input. Whether the phrase is literal or idiomatic, we must identify which image is closest to that meaning and rank the images accordingly. So, in this work, we aim to bridge this gap by analyzing how images convey idiomatic knowledge and investigating the relationship between visual and linguistic representations of idioms. To address this task, we propose an architecture composed of two expert models for the English language: one dedicated to processing idiomatic sentences and the other to handling sentences in their literal sense. We first classify the phrase in the sentence, and then the corresponding expert ranks images based on their specialized training. Further details about the architecture are provided in Section 4.

2 Related Work

BERT is a deep learning model introduced by (Devlin et al., 2019) that has revolutionized natural language processing (NLP) by leveraging a bidirectional Transformer architecture (Vaswani et al., 2017). Unlike traditional language models that process text sequentially, BERT captures context from both left and right directions, allowing it to understand the meaning of words in relation to their surroundings. Pre-trained on vast amounts of text using masked language modeling and next-sentence prediction tasks, BERT has demonstrated state-of-the-art performance on various NLP benchmarks.

CLIP is a multi-modal model developed by OpenAI (Radford et al., 2021) that learns to associate images with textual descriptions using contrastive learning. It consists of two separate encoders: a Transformer architecture (Vaswani et al., 2017) for processing text and a Vision Transformer (ViT) (Dosovitskiy et al., 2021) for encoding images. These encoders project their respective modalities

Proceedings of the The 19th International Workshop on Semantic Evaluation (SemEval-2025), pages 2270–2275

into a shared embedding space, where contrastive learning aligns visual and linguistic representations. Trained on diverse image-text pairs, CLIP enables zero-shot classification and retrieval even without task-specific fine-tuning, showcasing broad generalization across domains.

SemEval-2022 Task 2 (Tayyar Madabushi et al., 2022) focuses on multilingual idioms in three languages: English, Portuguese, and Galician. This task is divided into two subtasks: Subtask A evaluates a language model's ability to identify idiomatic expressions, while Subtask B assesses how effectively a model generates sentence representations containing idioms. Subtask A includes two evaluation settings: Zero-Shot and One-Shot, whereas Subtask B includes Pre-Training and Fine-Tuning settings. The dataset used in SemEval-2022 Task 2 is an extension of the one introduced by (Tayyar Madabushi et al., 2021). It comprises 8,683 entries across the three languages (English: 5,352, Portuguese: 2,555, Galician: 776). For Subtask A, multilingual BERT served as the baseline model, and for Subtask B, the approach involved introducing single tokens for each multiword expression (MWE) in the dataset.

(Phelps et al., 2024) investigates the capacity of LLMs to comprehend idioms. The study suggests that LLMs perform worse than fine-tuned encoder-only models on these tasks. However, it also observes that performance in idiomaticity detection improves as the model size increases.

3 Dataset

The AdMIRe dataset (Pickard et al., 2025) contains 200 data points, divided into four subsets: train, validation, test, and extended test, which accordingly have 70, 15, 15, and 100 data points. Each data point consists of a phrase, a sentence containing the phrase, five images, and corresponding captions. Additionally, each data point is annotated with a label indicating whether the phrase is used idiomatically in the sentence. The dataset also provides the expected ranking order of the images, representing the ground truth for their relevance to the phrase in the sentence.

Figure 1 shows a sample data point. The phrase for this data point is "open book," used idiomatically in the sentence, which means a person whose thoughts and feelings are easy to know. As you see, two of the images are close to literal meaning, two of them are close to idiomatic meaning, and there is an image that is completely different from the phrase and sentence, so the ranking must be "B E C A D". The labeled dataset is publicly available at this link.

For our idiom detection and representation models, we use the English subset of the AStitchInLanguageModels dataset (Tayyar Madabushi et al., 2021), which contains 4,645 examples from 223 different phrases. Each instance represents either the idiomatic or literal meaning of the phrase. Additionally, the dataset provides a literal meaning for each phrase, while phrases with idiomatic examples include 1 to 3 non-literal (idiomatic) meanings. The dataset is available at this link.

4 System Overview

Our system consists of three experts and works as follows. First, the BERT-based classifier receives the sentence and phrase as input to determine whether the phrase is used in its idiomatic or literal sense. If the phrase is used idiomatically, the idiomatic expert takes the sentence, phrase, image, and its caption to calculate a relevance score for each image and ranks them accordingly. If the phrase is used literally, the literal expert follows the same process to compute relevance scores and rank the images based on their alignment with the literal meaning of the phrase. The overview of our system is shown in Figure 2. We will see the details of each expert in the following.

4.1 Classifier Expert

We fine-tuned a BERT model for idiom classification on the AStitchInLanguageModels and AdMIRe datasets to serve as our classifier expert. The model takes a phrase and a sentence as input, separated by the [SEP] token (i.e., "phrase [SEP] sentence"). The [CLS] token embedding is extracted and passed through a projection layer that maps it to a logit for classification. During the training phase of the entire system, this classifier is not used since the true labels are available. Instead, the classifier is only employed at inference time when the labels are unknown.

4.2 Idiomatic Expert

Once a data unit is identified as a term, we use our term specifier to score its images. This expert consists of 4 components. The first component is a BERT model fine-tuned to take a sentence and the phrase and translate it into an embedding space that

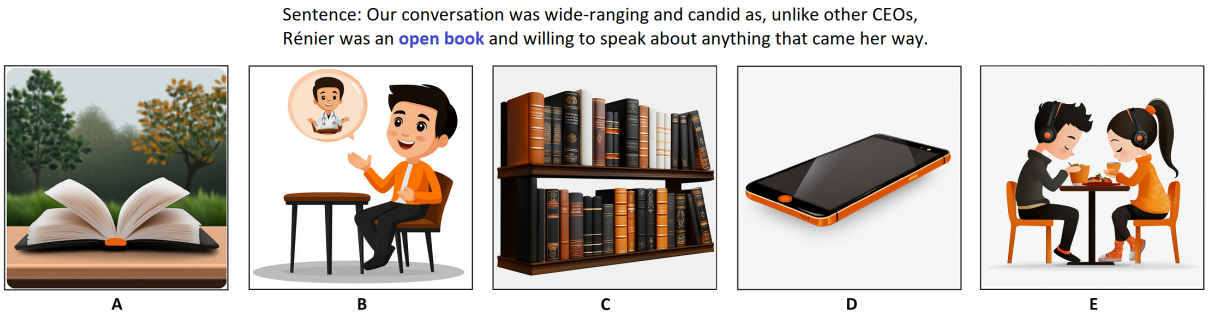

Figure 1: The images of a sample data point in the AdMIRe dataset, where the related phrase is "open book" and the sentence is "Our conversation was wide-ranging and candid, as, unlike other CEOs, Rénier was an open book and willing to speak about anything that came her way."

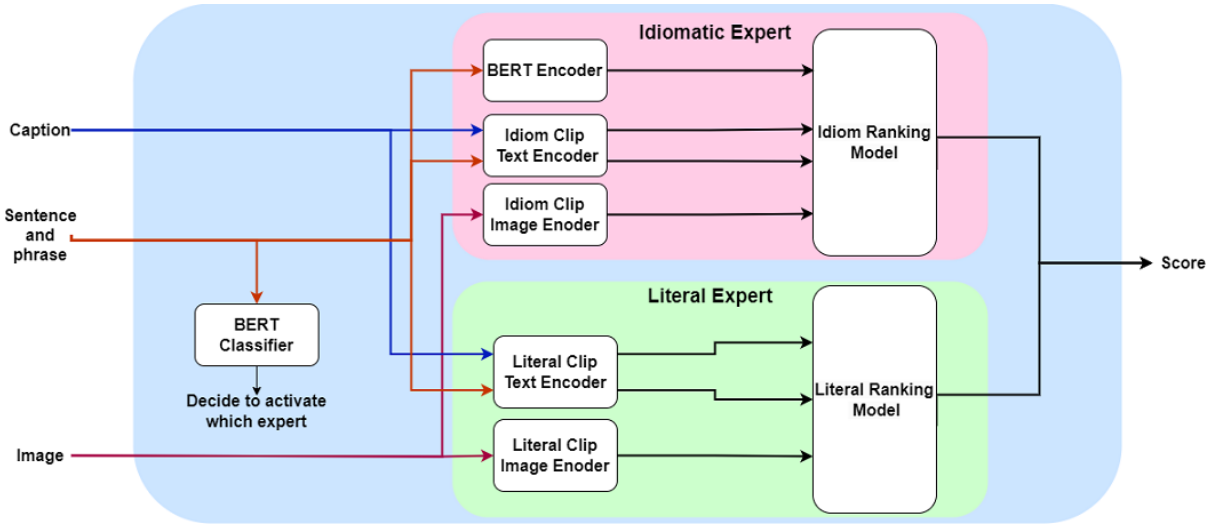

Figure 2: The first expert (BERT classifier) activates the idiomatic expert or literal expert, respectively, based on whether the phrase in the sentence has an idiomatic meaning or a literal meaning, so that the selected expert can assign a score by taking the sentence, image, and caption

has a good representation of the idiomatic meaning of that phrase. The details are discussed in Section 4.2.1.

The second and third components are the text encoder and image encoder of a pre-trained CLIP model. The text encoder maps sentences and captions into the embedding space, while the image encoder performs the same transformation for images, aligning both modalities in a shared space. Due to the limited size of our training data, one of the primary challenges is overfitting. To mitigate this, we simplified the model during training by freezing the image encoder and fine-tuning only the last two layers of the text encoder. Additionally, since many image captions exceeded the input length of the text encoder, we summarized them using the (Lewis et al., 2019) model, reducing their length to a maximum of 60 words.

The final component is a ranking model, implemented as a simple feed-forward neural network with one hidden layer. This network receives four 768-dimensional embeddings from the previous components and projects each embedding into a 128-dimensional hidden state using a fully connected layer followed by a ReLU activation function. The four hidden vectors are then concatenated and projected onto a single scalar value, representing the final similarity score for the input embeddings. The ranking model is trained simultaneously with the last two layers of the CLIP text encoder.

4.2.1 Bert Encoder

To improve the representation of phrase meanings in the BERT encoder, we incorporated additional loss components alongside the classification loss in the final objective function. Specifically, we provided the literal and idiomatic meaning of the phrase to a CLIP text encoder and extracted em-

Table 1: The rank-1 accuracy, rank difference, and classification accuracy for the entire system and each expert module independently

	Data	Rank-1 Acc.	Rank Diff.	Class Acc.
Entire System	Train	0.757	3.543	0.986
	Dev	0.667	4.933	0.800
	Test	0.650	5.807	0.933
	Ex-test	0.500	5.260	0.740
Idiom Expert	Train	0.744	3.641	0.974
	Dev	0.714	5.143	0.857
	Test	0.750	6.250	0.875
	Ex-test	0.510	4.939	0.826
Literal Expert	Train	0.774	3.419	1.000
	Dev	0.625	4.750	0.750
	Test	0.571	5.143	1.000
	Ex-test	0.480	5.548	0.667

beddings for each. Using cosine similarity, we introduced a contrastive loss that encourages the BERT encoder's output embedding to be closer to the embedding corresponding to the correct meaning (based on the ground truth label) and farther from the other.

4.3 Literal Expert

The literal expert architecture is very similar to the idiomatic expert architecture, with the main difference being that the BERT encoder is not used. This is because the task is simpler. The CLIP model has seen most expressions in their literal sense during its pre-training and does not require additional semantic information. As a result, the ranking model receives three inputs instead of four.

5 Experimental Setup

The dataset was split into training, validation, and test sets. We applied basic pre-processing like tokenization and encoding using the Hugging Face Transformers library. The model was trained with the AdamW optimizer, using a learning rate of 3×10^{-6} and a weight decay of 1×10^{-4}, over 30 epochs with a batch size of 4. To keep the results consistent, we set a fixed random seed. The training was done using PyTorch on a P100 GPU.

6 Results and Analysis

To evaluate the system during inference, we employ our BERT classifier for classification, where its errors directly impact the final output. The evaluation consists of two types of tests: one on the entire system and another on each expert module independently. For example, to measure the performance of the idiom expert, we input only data labeled with the idiom attribute into the system. The evaluation metrics include Rank-1 Accuracy and Rank Difference, defined as the absolute difference between the predicted and ground truth rankings. The results of both the overall system and individual expert modules are presented in Table 1. Further analyses are provided in the following sections.

6.1 Remove captions and use cosine for literal expert

In one of our experiments, we excluded the image captions and modified the literal expert module to compute the image score by measuring the cosine similarity between the image encoder's and text encoder's output embeddings of CLIP. This system was submitted to Task 1 of SemEval-2025, achieving 87% rank-1 accuracy on the test set and 48% on the extended test set.

As shown in Table 2, the baseline system (using cosine similarity without captions) achieves higher rank-1 accuracy on the test set compared to our proposed system. However, our system performs better on the extended test set, which contains a larger and more diverse set of samples. This suggests that our final system generalizes better to broader, unseen data compared to the baseline.

Table 2: Rank-1 accuracy for the baseline (without captions and cosine similarity for literals) and the proposed system on test and extended test sets.

	Data	Rank-1 Acc.
Baseline	Test	87%
	Ex-test	48%
Proposed System	Test	65%
	Ex-test	50%

6.2 Different Losses

One of the key factors for achieving better learning and generalization is selecting an appropriate loss function. Therefore, we train the models using different loss functions, including Pairwise Hinge, Listwise Softmax, and Top-1 Hinge. The details of these loss functions are provided in Appendix A, and the results of these experiments are presented

in Figure 3. As you can see, the pairwise loss function achieves higher accuracy compared to other loss functions.

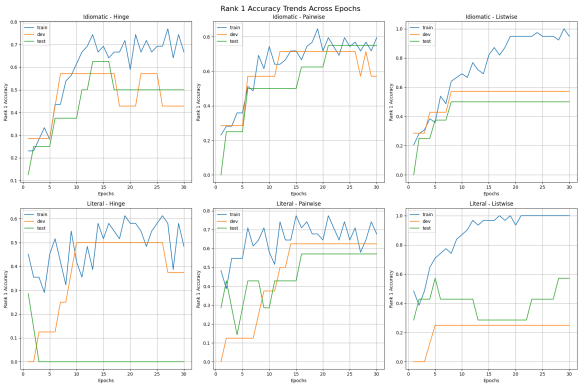

Figure 3: Rank 1 accuracy trends over epochs for different ranking loss functions on idiomatic and literal data. The results show that the pairwise loss function achieves higher accuracy compared to other loss functions, indicating better ranking performance.

6.3 Analyze impact of BERT encoder

One of the key components of the Idiom Expert model is its BERT encoder, which plays a crucial role in generating high-quality representations of idioms. In this experiment, we evaluated its contribution by removing the encoder and analyzing its impact on the model's performance. As you can see in Figure 4, using BERT embeddings helps our model achieve better performance and generalization.

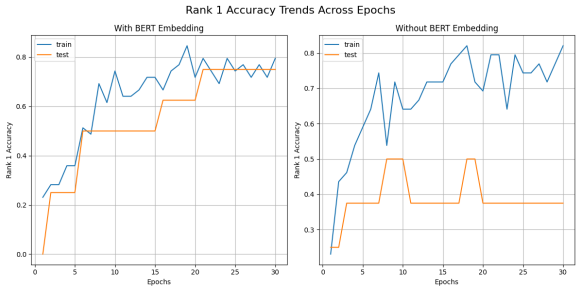

Figure 4: Rank 1 accuracy trends over epochs for idiomatic data. The left graph shows results using BERT embeddings as a feature of the ranking model, and the right one without using these embeddings. The BERT model helps the model generalize better and improves performance.

7 Conclusion

In this paper, we propose a multi-expert architecture to rank images based on whether a given phrase is used as an idiom or a literal expression in the accompanying sentence. By leveraging datasets from idiom detection tasks in the text domain, we successfully transfer their knowledge to the image space. Additionally, we explore various loss functions to identify the most effective one for the ranking task, demonstrating the impact of each component through ablation studies that remove different parts of the architecture.

For future work, these datasets open up exciting possibilities, such as generating images for idiomatic expressions, converting idiom-related images into textual descriptions, and other multimodal tasks. Moreover, new image-based datasets can be constructed from existing text datasets, facilitating the development of more robust models and addressing increasingly complex challenges in the intersection of language and vision.

References

Christos Baziotis, Prashant Mathur, and Eva Hasler. 2023. Automatic evaluation and analysis of idioms in neural machine translation. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 3682–3700, Dubrovnik, Croatia. Association for Computational Linguistics.

Verna Dankers, Christopher Lucas, and Ivan Titov. 2022. Can transformer be too compositional? analysing idiom processing in neural machine translation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3608–3626, Dublin, Ireland. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Sundesh Donthi, Maximilian Spencer, Om B. Patel, Joon Young Doh, Eid Rodan, Kevin Zhu, and Sean O'Brien. 2025. Improving LLM abilities in idiomatic translation. In *Proceedings of the First Workshop on Language Models for Low-Resource Languages*, pages 175–181, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An image

is worth 16x16 words: Transformers for image recognition at scale. *Preprint*, arXiv:2010.11929.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2019. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *CoRR*, abs/1910.13461.

Dylan Phelps, Thomas Pickard, Maggie Mi, Edward Gow-Smith, and Aline Villavicencio. 2024. Sign of the times: Evaluating the use of large language models for idiomaticity detection. In *Proceedings of the Joint Workshop on Multiword Expressions and Universal Dependencies (MWE-UD) @ LREC-COLING 2024*, pages 178–187, Torino, Italia. ELRA and ICCL.

Thomas Pickard, Aline Villavicencio, Maggie Mi, Wei He, Dylan Phelps, and Marco Idiart. 2025. Semeval-2025 task 1: Admire - advancing multimodal idiomaticity representation. In *Proceedings of the 19th International Workshop on Semantic Evaluations (SemEval-2025)*, Vienna, Austria. Association for Computational Linguistics.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR.

Ivan A. Sag, Timothy Baldwin, Francis Bond, Ann Copestake, and Dan Flickinger. 2002. Multiword expressions: A pain in the neck for nlp. In *Computational Linguistics and Intelligent Text Processing*, pages 1–15, Berlin, Heidelberg. Springer Berlin Heidelberg.

Bashar M. A. Tahayna, Ramesh Kumar Ayyasamy, and Rehan Akbar. 2022. Automatic sentiment annotation of idiomatic expressions for sentiment analysis task. *IEEE Access*, 10:122234–122242.

Harish Tayyar Madabushi, Edward Gow-Smith, Marcos Garcia, Carolina Scarton, Marco Idiart, and Aline Villavicencio. 2022. SemEval-2022 task 2: Multilingual idiomaticity detection and sentence embedding. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 107–121, Seattle, United States. Association for Computational Linguistics.

Harish Tayyar Madabushi, Edward Gow-Smith, Carolina Scarton, and Aline Villavicencio. 2021. AStitchInLanguageModels: Dataset and methods for the exploration of idiomaticity in pre-trained language models. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3464–3477, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

A Loss Functions

A.1 Pairwise Hinge Loss

This loss ensures that a higher-ranked item has a greater predicted score than a lower-ranked one by a margin m, penalizing violations:

$$\mathcal{L}_{\text{pairwise}} = \frac{1}{N} \sum_{i,j} \max(0, m - (s_i - s_j)) \cdot \text{sign}(r_j - r_i)$$

where s are predicted scores, r are ground truth ranks, and N is the number of item pairs.

A.2 Listwise Softmax Loss

This loss applies softmax normalization on ranking scores and minimizes cross-entropy to align predicted and true rankings:

$$\mathcal{L}_{\text{listwise}} = -\sum_{j} p_j \log \hat{p}_j$$

where $p_j = \frac{e^{-r_j}}{\sum_k e^{-r_k}}$ is the true rank distribution and $\hat{p}_j$ is the softmax-normalized prediction.

A.3 Top-1 Hinge Loss

This loss maximizes the margin between the top-ranked item and others while suppressing overall scores:

$$\mathcal{L}_{\text{top1}} = \sum_{j \neq \text{top1}} \max(0, m + s_j - s_{\text{top1}}) - s_{\text{top1}} + \alpha \sum_{j} s_j$$

where s_{top1} is the score of the most relevant item, and α controls non-top-1 suppression.