Advancing Multimodal Idiomaticity

Representation (AdMIRe)

Ziad Moataz Samy 211001556

Michael Wagdy Wahib 211001776

Mohamed Hossam Hassan 211000405

Mohamed Abdelazim 211001926

AIS411: Natural Language Processing-2024FALL

Nile University

Supervised by: Dr. Ensaf Hussien

Contents

[Introduction 3](#_Toc186336681)

[Related Work 5](#_Toc186336682)

[Methodology 8](#_Toc186336683)

[Subtask A : Image Ranking for Idiomatic Nominal Compounds (NC) in Context 8](#_Toc186336684)

[2. Data Preparation 8](#_Toc186336685)

[3. Dataset Handling 9](#_Toc186336686)

[4. Multimodal Model Design 9](#_Toc186336687)

[5. Model Training 10](#_Toc186336688)

[6. Evaluation Metrics 10](#_Toc186336689)

[7. Model Evaluation 11](#_Toc186336690)

[8. Results and Observations 11](#_Toc186336691)

[2.2 Subtask B: Image Sequence Prediction 11](#_Toc186336692)

[Image Preprocessing: 11](#_Toc186336693)

[Captions Processing: 12](#_Toc186336694)

[Dynamic Image Filtering: 12](#_Toc186336695)

[Model Architecture 12](#_Toc186336696)

[Feature Extraction 12](#_Toc186336697)

[Similarity Measurement 13](#_Toc186336698)

[Model Training 13](#_Toc186336699)

[Training Procedure 14](#_Toc186336700)

[Evaluation Metrics 14](#_Toc186336701)

[Visualization and Testing 14](#_Toc186336702)

[Evaluation Pipeline 15](#_Toc186336703)

[Results and Discussion 15](#_Toc186336704)

[Conclusion and future work 16](#_Toc186336705)

[References 17](#_Toc186336706)

**Abstract**

This has been made possible by remarkable progress in deep learning for AI and complex problem-solving seems to have leaped a step further in the more general direction of NLP, computer vision, and multimodal learning. This paper addresses two intertwined tasks that reflect the growing need for both powerful unimodal and multimodal AI models: Subtask A covers sequence prediction in a textual context, while Subtask B integrates textual and visual data to predict the next image in a sequence. Each subtask represents different but complementary challenges that reflect both the versatility and depth of modern AI systems.

Subtask A: Predictive Modeling of Sequential Text Data-Subtask A is concerned with the predictive modeling of sequential text data, a task at the center of many different applications which range from dynamic forecasting, contextual reasoning to generative systems. Our proposed approach leverages the current state-of-the-art transformer-based architectures to model complicated dependencies in textual sequences with a view to making contextually well-grounded predictions. Fine-tuning on domain-specific data enabled capturing peculiarities of the task at very high performance with all standard metrics.

Subtask B requires a model to predict an image that comes next in a sequence, given previous images and their captions. Here, the model has to learn the pattern of visual sequences, interpret the text description, and then select the most probable candidate out of the choices. This paper demonstrates that fine-tuning CLIP on task-specific data is a very effective way to handle this multimodal prediction challenge.

# Introduction

With the fast evolution of AI and ML, there is a change in the working of various fields, ranging from NLP and computer vision to multi-modal applications that integrate several data modalities. The advancement in these fields has made it possible to solve certain complicated problems which were considered unsolvable in the past. A particular set of such problems encompasses those that require understanding, synthesizing, and predicting patterns from diverse sources of data, including but not limited to textual descriptions and visual inputs. It discusses two such tasks: Subtask A, which is related to sequence prediction in a textual context, and Subtask B, which integrates both textual and visual modalities to predict the next image in a sequence.

Subtask A uses the NLP capabilities of modern transformer-based advanced models. This task deals with a sequence prediction problem where, according to the patterns the model learns from the history, it must predict an element in the sequence. These kinds of problems find applications in recommendation systems, generative models, and contextual reasoning. The recent revolution of NLP models, especially those using transformer architecture, such as BERT and GPT and their derivatives, has brought tremendous gain in capability regarding the modelling of contextual and sequential dependencies in textual data. The approach here relies on robust transformer models fine-tuned on domain-specific data to perform highly accurately with reliable results.

While subtask A is more of a traditional challenge for a model, much harder subtask B adds visual data on top of text. Such multimodal tasks are at the top of the current agenda of AI research: here, models must extract meaningful representations from both text and images and then integrate those representations to make predictions. Such tasks are particularly relevant in domains such as e-commerce, medical imaging, and digital content creation, where the relationship between textual descriptions and visual elements needs to be understood. In multimodal AI research, CLIP models have recently gained significant importance since they learn rich joint embeddings of images and text, making them very appropriate for tasks that involve both modalities.

Subtask B requires a model to predict an image that comes next in a sequence, given previous images and their captions. Here, the model must learn the pattern of visual sequences, interpret the text description, and then select the most probable candidate out of the choices. This paper demonstrates that fine-tuning CLIP-on task-specific data is a very effective way to handle this multimodal prediction challenge.

Each subtask requires the use of datasets that incorporate a range of real-world complexities and challenges, such as the variability in text descriptions, noisiness of data, and variety of visual inputs. Whereas Subtask A draws on textual datasets, each formatted to focus on sequential relations, Subtask B combines textual and visual data and thus calls for novel pre-processing and fine-tuning strategies in order to effectively train models.

# Related Work

The emergence of multimodal learning has driven the benefits for image-text and text-image retrieval systems that would make the machines capable of viewing and processing visual and textual information. Amongst these progress developments, CLIP model performance stood above by OpenAI in zero-shot classification, visual question answering, and multimodal retrieval tasks. In this paper, the authors present the CLIP model, which integrates vision and language into a shared space with embeddings that are far more powerful for tasks needing semantic alignments of images and texts.

A few works have extended the capability of CLIP by adapting it to domain-specific tasks. Radford et al. proposed a CLIP architecture that is designed to jointly encode images and text by a transformer-based model to allow natural language supervision for any visual learning task. Based on the base provided, several works have been performed to adapt CLIP to fine-tuning on specialized applications, including medical image analysis, product recommendation systems, and video captioning.

One line of research has been to enhance CLIP's performance by fine-tuning techniques and task-specific augmentations. For example, Liu et al. (2022) demonstrated how prompt engineering and fine-tuning strategies could be applied to improve the adaptability of CLIP to downstream tasks. In the same vein, Jia et al. (2021) pointed out the importance of multimodal pretraining to be able to make state-of-the-art performance in image-text matching tasks.

Few previous works incorporate RNNs and transformers to model temporal dependencies for the task of sequential image prediction and ranking. Wang et al. (2020) proposed frameworks which are sequence-aware; the sequences of images are thus aligned with coherent narratives. In contrast, we make use of the pre-trained CLIP embeddings to encode both a sequence of images and candidate images, including textual description complementary features.

Some of the previous works investigated image similarity measures and feature fusion. The conventional methods have relied on hand-engineered features and statistical similarities, their modern variants based on deep neural networks for feature extraction. To mention a few, especially, Kim et al. have combined visual embeddings with textual embeddings to develop cross-modal matching tasks. Our method borrows from the use of textual features combined with averaged sequence embeddings in the ranking of candidate images by their contextual relevance.

In addition, several visualization techniques were used to interpret the model predictions by assessing how well the features in an image align with a given textual description. Selvaraju et al. introduced a technique for visualizing the importance of features known as Grad-CAM in the year 2017, while recent methods incorporate the use of attention maps to show relevant large regions in images.

The present work thus follows these studies, using a CLIP model for a multimodal ranking task but extends the literature by introducing sequence-aware processing, feature aggregation mechanisms, and candidate ranking mechanisms designed particularly for the AdMIRe task. Finally, the experimental design also insists on interpretability: it visualizes prediction and analyzes errors.

Experimental Results

To evaluate the proposed approach, we implemented two sets of experiments corresponding to the Subtasks A and B of the AdMIRe competition, respectively. These experiments targeted the ranking of images w.r.t. idiomatic expressions and contextual information by means of the CLIP model capabilities.

Error Analysis

Error patterns unveiled the following important shortcomings:

1. Too much dependence on the textual captions when no other visual cues were around.
2. Difficulty handling abstract concepts or idiomatic expressions requiring cultural knowledge.
3. Sometimes, failure to make fine-grained visual distinctions leads to predictions biased toward semantically similar distractors.

# Methodology

## Subtask A : Image Ranking for Idiomatic Nominal Compounds (NC) in Context

### 2. Data Preparation

The data preparation phase involves loading and organizing the data for effective model training and evaluation.

#### 2.1. Data Loading

* The dataset is imported using the Pandas library, enabling structured handling of tabular data.
* Textual data, including sentences and idioms, is read from CSV files, while corresponding image files are organized in directories.
* Target rankings for images are extracted from labeled data to evaluate performance.

#### 2.2. Image Preprocessing

* Image preprocessing utilizes PyTorch's transforms module to standardize input dimensions (224x224) and normalize pixel values.
* Transformations include resizing, tensor conversion, and normalization based on ImageNet statistics to ensure compatibility with pretrained ResNet-50.

#### 2.3. Text Preprocessing

* Sentences are tokenized using BERT's tokenizer, generating input IDs and attention masks required for processing textual inputs.
* Tokenized sequences are padded and truncated to a maximum length of 128 tokens to maintain uniformity.

### 3. Dataset Handling

A custom PyTorch dataset class (IdiomImageDataset) is implemented to handle multimodal inputs. Key functionalities include:

* Mapping idiomatic expressions to their associated images.
* Loading and preprocessing images and text for each data point.
* Pairing target rankings with image sets for supervised learning.

### 4. Multimodal Model Design

The proposed multimodal architecture integrates textual and visual modalities using pretrained models:

#### 4.1. Textual Feature Extraction

* A pretrained BERT model (bert-base-uncased) extracts contextual embeddings from input sentences.
* Sentence representations are obtained by averaging token embeddings.

A diagram of a process

Description automatically generated

#### 4.2. Visual Feature Extraction

* A ResNet-50 model extracts image features by removing its fully connected layer, yielding a 2048-dimensional feature vector per image.

A group of blue rectangular objects with black text

Description automatically generated

#### 4.3. Feature Fusion

* Text and image features are concatenated and passed through fully connected layers to learn joint representations.
* Dropout regularization (30%) is applied to mitigate overfitting.

#### 4.4. Ranking Prediction

* The final output layer predicts scores for ranking the five input images.

### 5. Model Training

#### 5.1. Data Splitting

* The dataset is split into training (80%) and testing (20%) sets to enable performance evaluation.
* Data loaders with batch size 32 are created for efficient processing.

#### 5.2. Loss Function and Optimizer

* Cross-entropy loss measures discrepancies between predicted and true rankings.
* The Adam optimizer is employed with a learning rate of 5e-5 for parameter updates.

#### 5.3. Training Procedure

* The model undergoes 50 epochs of training, with loss computed for each image's ranking.
* Backpropagation and gradient updates refine model parameters iteratively.

#### 5.4. Monitoring Progress

* Loss values are printed after each epoch, facilitating the identification of trends and convergence.

### 6. Evaluation Metrics

#### 6.1. Ranking Accuracy

* Predicted rankings are compared with true rankings based on Mean Reciprocal Rank (MRR).
* MRR evaluates how closely predicted ranks align with the ideal order, measuring precision at the top of the rankings.

#### 6.2. Statistical Measures

* Kendall's Tau correlation coefficient evaluates rank similarity, capturing ordinal relationships between predicted and actual rankings.
* Additional metrics like precision and recall are computed to assess overall performance.

### 7. Model Evaluation

The evaluation phase applies the trained model to the test set

* Images are processed in batches for inference.
* Predicted ranks are extracted and converted into ordered lists.
* True and predicted rankings are compared to compute MRR and Kendall's Tau scores, providing quantitative insights into model performance.

### 8. Results and Observations

Training results show consistent loss reduction across epochs, indicating successful learning. Key observations include:

* Final MRR of 0.5278 (52.78%) demonstrates moderate ranking accuracy.
* Performance improves steadily, with lower loss values over time.
* Errors predominantly occur in ranks with visually ambiguous images, suggesting areas for further refinement.

## 2.2 Subtask B: Image Sequence Prediction

### Image Preprocessing:

* + Images are loaded using the Python Imaging Library (PIL) and resized to match the input requirements of the CLIP model (224x224 pixels).
  + Images are converted into RGB format to maintain consistency and ensure compatibility with the CLIP model.
  + Processed images are transformed into tensor format using the CLIPProcessor and normalized to match the pre-trained CLIP input distribution.

### Captions Processing:

* + Captions are concatenated from multiple columns to provide contextual descriptions.
  + Text inputs are tokenized and padded/truncated to meet the maximum token limit of the CLIP model (77 tokens).

### Dynamic Image Filtering:

* + Candidate images are dynamically selected based on sequence order, focusing on the last three images as primary candidates.

### Model Architecture

Pre-trained Model Selection

A pre-trained Contrastive Language-Image Pretraining (CLIP) model, specifically the "openai/clip-vit-base-patch32" variant, is employed. This model is chosen for its ability to process both textual and visual modalities, providing embeddings that align semantically across the two domains.

A diagram of a diagram

Description automatically generated

### Feature Extraction

#### Sequence Image Features:

* + Sequence images are batch-processed and passed through the visual encoder of the CLIP model.
  + Image features are normalized and averaged to create a unified embedding for the sequence.

#### Candidate Image Features:

* + Candidate images are processed in parallel and embedded into feature vectors.
  + Normalization is applied to ensure consistent magnitude and scaling for similarity comparisons.

#### Text Features:

* + Captions are tokenized, embedded using the CLIP text encoder, and normalized to align with image embeddings.

#### Fusion of Features:

* + The average embedding of sequence images is combined with the text embedding to generate a multimodal representation.

### Similarity Measurement

The cosine similarity metric is computed between the combined embedding and each candidate image embedding. Predictions are made by identifying the candidate with the highest similarity score.

### Model Training

Training Setup

#### Hardware Configuration:

* + The model runs on GPU-enabled Google Colab environments when available; otherwise, CPU fallback is enabled.

#### Hyperparameters:

* + Learning Rate: 5e-5
  + Batch Size: 4
  + Epochs: 10

#### Optimizer:

* + AdamW optimizer is employed for parameter updates, chosen for its adaptive learning rate and weight decay regularization.

### Training Procedure

#### Gradient Anomaly Detection:

* + Enabled to identify and resolve issues related to gradient flow during backpropagation.

#### Loss Function:

* + Cross-entropy loss is computed using similarity scores as logits.
  + Ground truth labels are dynamically assigned to index 0 within candidate images.

#### Backpropagation:

* + Loss gradients are calculated and weights are updated iteratively.

#### Performance Monitoring:

* + Epoch-wise losses and accuracy metrics are logged to track performance trends.

### Evaluation Metrics

* Accuracy:
  + Computed as the proportion of correctly predicted images relative to the total number of predictions.
* Loss:
  + Monitored to ensure model convergence and detect overfitting trends.

### Visualization and Testing

#### Model Testing

The evaluation pipeline uses unseen test data loaded from Google Drive. The same preprocessing and embedding pipeline is applied, ensuring consistent data representation.

#### Visualization Pipeline

1. Display Functionality:
   * Sequential images, predicted images, and ground truth images are displayed side-by-side for qualitative evaluation.
2. Error Analysis:
   * Samples with mismatched predictions are visualized to analyze failure cases and improve model robustness.

# Evaluation Pipeline

1. Feature Extraction in Testing Mode:
   * Sequence and candidate images are processed identically to the training phase.
2. Prediction and Visualization:
   * Predicted images are compared visually against ground truth images.

# Results and Discussion

The proposed multimodal ranking framework effectively merges the image and text data in predicting the ranking of the images with respect to a given idiom. The key component of the proposed system involves a fine-tuned BERT model for acquiring text embeddings, a pre-trained ResNet50 in extracting features from images, fused into making accurate rankings by means of a custom neural architecture.

Performance on Training One can observe that the loss decreases with an increase in the number of training epochs, which in other terms is great learning. Early epochs from 1 to 10 showed great improvements, where at the beginning, the Loss is ~8.05 reduced to ~5.80; having converged at ~ 0.14 at an epoch of 50, the model had surely become robust enough for learning complex multimodal associations.

Model Behaviour: Introduction of BERT for Sentence embeddings is important, which hitherto provided the model with the semantic meaning in textual idioms. Meanwhile, ResNet50 modified with an identity layer ensured a proper probe on important features to get captured along with providing fair probing feature and thereby assists in finding more subtle differences of a picture and help model make enhancements over the ranking carrying weightage to explain picturing related importance regarding a context.

# Conclusion and future work

The final result that this paper tries to indicate is a new rank model for idiomatic image associations in a multimodal way. It effectively leverages pre-trained BERT comprehension with ResNet50 image analysis in the ranking prediction task. Smoothly decreasing loss values during training underpin that the combined architecture learned complex relationships between text and images, strongly reinforcing the effectiveness of using a multimodal approach in this work.

Further verification may be done with other models such as RoBERTa or GPT-based embeddings that are said to catch the semantics much better in a lot richer textual embedding. Besides, on the image side, more active models like Vision Transformers might be applied for feature extraction to improve the performance concerning image ranking. These datasets need to be extended and diversified with regards to achieving more robustness and generalization.

# References

Shen, S., Li, L.H., Tan, H., Bansal, M., Rohrbach, A., Chang, K.W., Yao, Z., & Keutzer, K. (2021). How much can CLIP benefit vision-and-language tasks? arXiv preprint arXiv:2107.06383. <https://doi.org/10.48550/arXiv.2107.06383>

Lin, Z., Geng, S., Zhang, R., Gao, P., De Melo, G., Wang, X., Dai, J., Qiao, Y., & Li, H. (2022, October). Frozen CLIP models are efficient video learners. In European Conference on Computer Vision (pp. 388-404). Cham: Springer Nature Switzerland. <https://link.springer.com/chapter/10.1007/978-3-031-19833-5_23>

Wang, Z., Liu, W., He, Q., Wu, X., & Yi, Z. (2022). CLIP-GEN: Language-free training of a text-to-image generator with CLIP. arXiv preprint arXiv:2203.00386. <https://doi.org/10.48550/arXiv.2203.00386>

Luo, H., Ji, L., Zhong, M., Chen, Y., Lei, W., Duan, N., & Li, T. (2022). CLIP4CLIP: An empirical study of CLIP for end-to-end video clip retrieval and captioning. Neurocomputing, 508, 293-304. <https://doi.org/10.1016/j.neucom.2022.07.028>

Alammary, A.S. (2022). BERT models for Arabic text classification: A systematic review. Applied Sciences, 12(11), 5720. <https://doi.org/10.3390/app12115720>

Tsai, H., Riesa, J., Johnson, M., Arivazhagan, N., Li, X., & Archer, A. (2019). Small and practical BERT models for sequence labeling. arXiv preprint arXiv:1909.00100. <https://doi.org/10.48550/arXiv.1909.00100>

Wang, Z., Ng, P., Ma, X., Nallapati, R., & Xiang, B. (2019). Multi-passage BERT: A globally normalized BERT model for open-domain question answering. arXiv preprint arXiv:1908.08167. <https://doi.org/10.48550/arXiv.1908.08167>

Singh, M., Jakhar, A.K., & Pandey, S. (2021). Sentiment analysis on the impact of coronavirus in social life using the BERT model. Social Network Analysis and Mining, 11(1), 33. <https://link.springer.com/article/10.1007/s13278-021-00737-z>

Markowski, M., Yearley, C., & Bower, H. (2022). Collaborative Learning in Practice (CLiP) in a London maternity ward-a qualitative pilot study. Midwifery, 111, 103360. <https://doi.org/10.1016/j.midw.2022.103360>

He, W., Vieira, T. K., Garcia, M., Scarton, C., Idiart, M., & Villavicencio, A. (2024). Investigating Idiomaticity in Word Representations. Computational Linguistics, 1–48. <https://doi.org/10.1162/coli_a_00546>