applied sciences

MDPI

Article

Thangka Image—Text Matching Based on Adaptive Pooling Layer and Improved Transformer

Kaijie Wang, Tiejun Wang *, Xiaoran Guo, Kui Xu and Jiao Wu

Key Laboratory of China's Ethnic Languages and Information Technology of Ministry of Education, Northwest Minzu University, Lanzhou 730030, China; foxnotail@foxmail.com (K.W.); guoxr1982@163.com (X.G.)
* Correspondence: wtj@mail.lzjtu.cn; Tel.: +86-136-7946-5982

Abstract: Image–text matching is a research hotspot in the multimodal task of integrating image and text processing. In order to solve the difficult problem of associating image and text data in the multimodal knowledge graph of Thangka, we propose an image and text matching method based on the Visual Semantic Embedding (VSE) model. The method introduces an adaptive pooling layer to improve the feature extraction capability of semantic associations between Thangka images and texts. We also improved the traditional Transformer architecture by combining bidirectional residual concatenation and mask attention mechanisms to improve the stability of the matching process and the ability to extract semantic information. In addition, we designed a multi-granularity tag alignment module that maps global and local features of images and text into a common coding space, leveraging inter- and intra-modal semantic associations to improve image and text accuracy. Comparative experiments on the Thangka dataset show that our method achieves significant improvements compared to the VSE baseline method. Specifically, our method improves the recall by 9.4% and 10.5% for image-matching text and text-matching images, respectively. Furthermore, without any large-scale corpus pre-training, our method outperforms all models without pre-training and outperforms two out of four pre-trained models on the Flickr30k public dataset. Also, the execution efficiency of our model is an order of magnitude higher than that of the pre-trained models, which highlights the superior performance and efficiency of our model in the image–text matching task.

Keywords: Thangka; image–text matching; adaptive pooling layer; bidirectional residual connection; masking attention mechanisms

Citation: Wang, K.; Wang, T.; Guo, X.; Xu, K.; Wu, J. Thangka Image—Text Matching Based on Adaptive Pooling Layer and Improved Transformer. *Appl. Sci.* **2024**, *14*, 807. https://doi.org/10.3390/app14020807

Academic Editor: Paolo Branchini

Received: 4 December 2023
Revised: 12 January 2024
Accepted: 16 January 2024
Published: 17 January 2024

1. Introduction

With the rapid development of multimedia and internet technologies, the form of information dissemination has shifted from single modality to encompassing cross-media data, including images, videos, texts, etc. The emergence of cross-media data has made multimodal information processing a focus of research in both academia and industry. The association and matching between images and texts are fundamental for tasks such as image–text generation, multimodal knowledge graph construction, and retrieval.

Early multimodal image–text matching methods predominantly utilized machine learning techniques to construct models. For instance, Hotelling [1] proposed Canonical Correlation Analysis (CCA) to measure the correlation and similarity between different modalities by constructing a shared semantic space. These methods often required the manual design of feature extractors and yielded suboptimal results when dealing with complex image–text matching problems.

Deep learning approaches, leverage multi-layer neural networks to automatically learn and extract features, eliminating the need for manual feature engineering and offering greater flexibility and adaptability. Additionally, deep learning models possess powerful representation capabilities, allowing them to capture complex relationships between images and texts and achieve better performance in image–text matching tasks. Visual Semantic

Embedding (VSE) [2] is a commonly used image–text matching method. It embeds images and texts for efficient cross-modal retrieval, and typically has the following steps. The image and text are extracted as features by separate visual and text encoders. These features are then projected into a joint embedding space and pooled to form fixed-length vectors. Then, the similarity calculation is utilized to measure the distance between instances and a suitable target is chosen for optimization. For example, Lee [3] proposed the Stacked Cross Attention Network (SCAN), which utilizes attention mechanisms to process features of images and texts separately and compute potential relationships between corresponding regions, achieving favorable results. However, this model overlooks potential semantic relationships in the context. Zhang [4] introduced the Context-Aware Attention Network (CAAN), which enhances the focus on potential local semantics in the context and effectively aggregates contextual information, capturing deeper intra-modal correlations and improving retrieval performance. Similarly, to enhance attention to semantic relationships, Chen [5] proposed the Iterative Matching with Recurrent Attention Memory (IMRAM) method, which iteratively updates cross-modal attention cores to explore fine-grained correspondences between images and texts. However, there are still problems, such as ignoring the relationship between text semantic features and insufficient acquisition of correlation between modalities.

With the rise of large language models in the field of BERT [6] and NLP, various large-scale pre-training models have appeared in the field of CV, which has also promoted the development of visual–text cross-modal pre-training models. Their usual mode is, first, pre-training on some set image and text tasks, and then fine-tuning on downstream tasks. Thanks to the training of large-scale data, their performance on downstream tasks far exceeds that of non-pre-trained models, and they have very good results in image and text retrieval and matching. For example, Jiasen [7] proposed a BERT architecture to learn the joint representation of images and text by integrating the feature extraction process, and achieved good results. Li [8] proposed to use of object labels as anchors to align image and language modalities in a shared semantic space. The model was pre-trained on a public corpus of 6.5 million image–text pairs, including graphs. It has achieved good results on multiple tasks, such as text matching. The multi-modal matching method based on pre-trained models can effectively solve the limitations of manual feature extractor design and the inability to handle complex image–text associations. However, it also has some shortcomings, such as the large training dataset required for the model. As well as problems such as long training cycles, low model efficiency, and lack of domain knowledge.

Thangka is an ancient Tibetan painting art form with rich color and complex backgrounds, which has high aesthetic value and artistic quality. In addition, the Thangka is known as the encyclopedia of Tibetan culture, and each Thangka image symbolizes deep religious and cultural connotations. Thangka image and text matching are crucial for bridging the semantic gap between images and text as well as achieving data intelligence tasks such as multimodal fusion, relational reasoning, and content generation. Meanwhile, due to the small amount of data and strong text–image correlation in the Thangka itself, the existing image-matching methods are unable to solve the problem of image matching in the Thangka domain better; so, a key unsolved problem is how to effectively extract image and text features from Thangka and capture fine-grained global and local correlations for image matching.

In order to solve the above challenges, this paper focuses on 30 kinds of Thangka with different themes, and proposes a new image–text matching method based on the VSE model, which combines adaptive feature aggregation and an improved Transformer [9]. This approach introduces three main improvements:

(1) The integration of adaptive pooling techniques in the model framework enhances the feature extraction capability for semantic correlations between Thangka images and texts.

(2) The traditional Transformer architecture is improved by incorporating bidirectional residual networks and masked attention mechanisms, increasing the stability of the matching process and the representation capability of semantic information.

(3) A multi-granularity label modal alignment module is designed to map global and local features of images and texts into a shared encoding space, fully exploring the semantic correlations between modalities and within modalities, thereby enhancing the precision of image–text matching.

2. Thangka Image–Text Matching Model

The proposed framework of the Thangka image–text matching model in this paper is shown in Figure 1. It consists of four main components: the image feature extraction module, the text feature extraction module, the encoder module, and the multi-granularity alignment module.

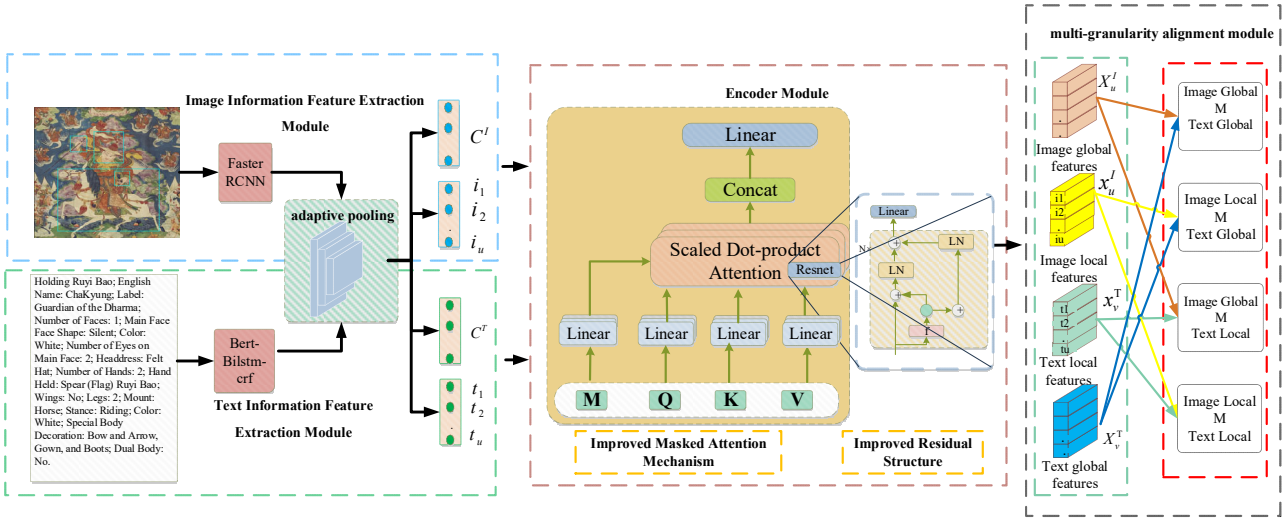

Figure 1. Model Framework Diagram.

For the imaging modality, considering the diverse target objects in Thangka images, the small pixel blocks occupied by each object, and the existence of pixel adhesion among different targets, we first use the top-down visual attention mechanism based on the Faster R-CNN [10] model to extract the target regions in Thangka images. Then, in the process of abstracting the feature matrix into feature vectors, adaptive pooling techniques are introduced, allowing the model to automatically learn the weight determination process. The subsequent visual and textual encoder modules learn and combine these features to generate fixed-size embedding features, which better extract the local feature vectors of the target objects in Thangka images and are represented as $E_u^I : \{i_1, i_2, i_3, \ldots, i_u\}$; i_u represents the u-th feature dimension of the vector. The global features are extracted using the feature extraction network of the trained Faster R-CNN, represented as C^I.

For the text modality, the given Thangka text is first converted into a vector representation. A pre-trained entity relation extraction model is used to extract (entity-relation-entity) triplets from the Thangka text data, which are then merged into text. The merged text and entity fragments are separately processed through a pre-trained bidirectional GRU [11] model, which introduces adaptive pooling layers to extract text features while preserving contextual semantic information as much as possible. This process yields local text feature vectors containing contextual semantic structures, represented as $E_u^T : \{t_1, t_2, t_3, \ldots, t_u\}$. The global feature vector for the text is extracted using the feature extraction module, represented as C^T.

After extracting the global and local features from both the image and text modalities, we use the encoder module to establish multi-granularity semantic associations between the two modalities. The encoder adopts an improved Transformer network

structure that combines bidirectional residuals and masked attention mechanisms. This structure solves potential gradient vanishing and model collapse issues and focuses on learning multi-granularity semantic information with similar structures between modalities. It outputs multi-granularity feature representations that combine information from different dimensions.

Finally, based on the obtained multi-granularity feature vectors, the modal alignment module calculates the losses for minimizing the correlation of local corresponding region features between positive samples, the correlation between regions and the whole, and maximizing the correlation of local corresponding region features between negative samples. This ensures that the learned region features have good discriminability, thereby guaranteeing the performance of Thangka image–text matching.

2.1. Thangka Image Information Feature Extraction Module

The Thangka image feature extraction module includes two parts: global features and local features. Inspired by models like SCAN [2] and SGRAF [12], this paper adopts a top-down visual attention mechanism based on the Faster R-CNN model to extract the local features $E_u^I : \{i_1, i_2, i_3, \ldots, i_u\}$ from the Thangka images. The global features are obtained by extracting features using the trained feature extraction network of Faster R-CNN, represented as C^I.

The specific implementation process is as follows: Firstly, the target objects in the Thangka image dataset are manually annotated and used to train a reliable and accurate object detection model. Then, the trained object detection model is used to obtain visual feature information and bounding box ranges of the Thangka images. Next, in the process of abstracting the feature matrix into feature vectors, an adaptive pooling layer is introduced to obtain local feature vectors with more key semantic information. At the same time, the entire image is used as input to obtain the global feature information of the image to enhance its semantic information. Finally, the feature attention information between each region and other regions in the image is encoded, and a fully connected layer is used to output the final image global feature vector C^I and local feature vectors E_u^I.

Considering the strong relevance and close entity mapping between Thangka images and text, in order to further capture the correlation between image features and text features, this paper, based on the work of Zhang [13], introduces an adaptive pooling layer to associate the local fine-grained information of Thangka images and text features.

The adaptive pooling technique combines the learning results of both the token layer and the embedding layer, allowing the model to automatically learn the weight determination process. Meanwhile, regularization methods are used to ensure that the abstract features of the image and text can be mapped to a similar space, so that the obtained image and text features can capture more key semantic information. The calculation process is shown in Figure 2.

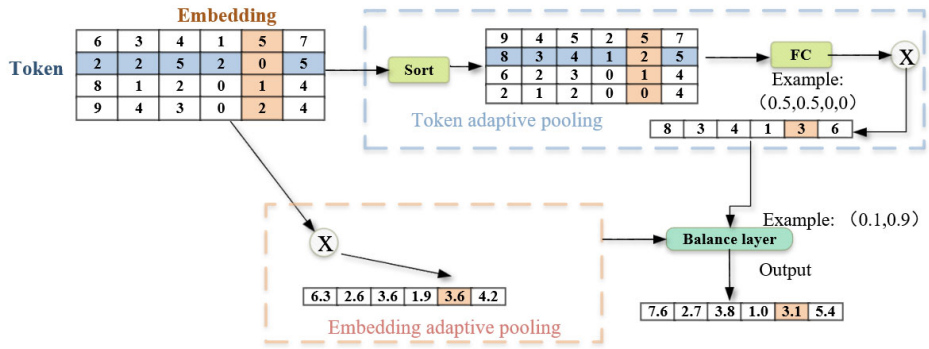

Figure 2. Adaptive pooling module.

In the token layer, following the "sort-weighted sum" paradigm of the simple pooling method, the feature matrix of the input matrix is first sorted along the embedding axis. Then,

a fully connected layer is set up to learn how to automatically weight each weight vector. In this way, the model can find the optimal combination of MeanPool and K-MaxPool in an adaptive manner, as shown in Equation (1):

$$\begin{cases} \{U_m\}_{m=1}^{M} = sort\left(\{t_m\}_{m=1}^{M}\right) \\ \theta = softmax\left(\{U_m\}_{m=1}^{M} W_{tok}\right) \\ t^{tok} = \sum_{m=1}^{M} \theta_m * U_m \end{cases} \tag{1}$$

Taking the blue row $\{t_m\}_{m=1}^{M}$ in Figure 2 as an example, the input matrix t is first sorted along the embedding axis using the sort function to obtain the sorted row data $\{t_m\}_{m=1}^{M}$. After normalization, a fully connected layer is established to learn how to automatically weigh each vector (brown rows in the matrix), resulting in the weight matrix θ_m. Then, weighting is performed to obtain the pooled vector t^{tok} in the token layer.

Considering only the weights obtained from the token layer, which have a similar distribution to MeanPool and K-MaxPool, a fully connected layer is also set up in the embedding layer to learn how to automatically weight each weight vector. This is shown in Equation (2).

$$\begin{cases} t^{emb} = \sum_{i=1}^{M} \delta_{ij} * t_{ij}\ , \forall j \\ \delta_{ij} = \frac{e^{t_{ij}}}{\sum_{i=1}^{M} e^{t_{ij}}}, \forall i \end{cases} \tag{2}$$

In the embedding layer pooling, each unsorted original vector t_{ij} is separately computed. The weight value for each vector is calculated using δ_{ij}, and then weighting is performed to obtain the pooled vector t^{emb} in the embedding layer.

The token and embedding vectors extract feature information in two dimensions. To merge them, two hyperparameters *w1* and *w2* are introduced. To avoid manually adjusting these hyperparameters, we use a trainable linear layer to learn the parameters, as shown in Equation (3):

$$t^{total} = w1\ t^{tok} + w2\ t^{emb} \tag{3}$$

In the experiment, text features are first extracted using the adaptive pooling layer described above, resulting in the text feature vector t_t^{total}. Then, using the trained pooling layer weights θ as the initial data for the adaptive pooling layer of the image, the adaptive pooling layer can integrate the potential feature information of both text and image as much as possible. Afterward, the token layer vector and embedding vector of the image are computed through the same process and combined to obtain the image feature vector t_i^{total}.

2.2. Thangka Text Information Feature Extraction Module

The Thangka text feature extraction consists of two parts: global features and local features. The local features $E_u^T : \{t_1, t_2, t_3, \ldots, t_u\}$ of Thangka text are extracted using a named entity recognition model. Then, the global feature vector $\boldsymbol{C^T}$ is extracted from the text using the feature extraction module of the model.

Extracting triplets from the Thangka text corpus is a complex task as a single sentence often contains multiple triplets with overlapping entities. For example, the sentence "The Buddha is the transliteration of the Sanskrit word "Buddha" and is commonly abbreviated as "Budda" in English. There are other transliterations such as "Buddho", "Buddhato", "Buddhata", "Buddhota", and "Buddhavacana". Additionally, there are translated terms like "Awakened One" and "Enlightened One" that represent the meaning of "Buddha". involves multiple entity relationships. There is a transliteration relationship between "Buddha" and "Budda" in Sanskrit to Latin. The entities "Buddho, Buddhato, Buddhata, Buddhota, Buddhavacana" are aliases of "Buddha", and there are also translated relationships.

To handle the complexity of overlapping entities and complex relationships in Thangka texts, a sequence labeling approach, following the research method of Guo [14], is adopted to annotate entities and relationships and obtain training data. Based on the annotated data, a named entity recognition model based on Bert-BiLSTM-Crf [15] is used to extract entity and relationship information from the text. The entity and relationship information obtained from the named entity recognition model and reorganized and concatenated into a triplet structure text. Word2Vec [16] word embedding model is used to represent the text as word vector features. An adaptive pooling layer is introduced, and the word vector features are fed into a bidirectional GRU [11] model to obtain contextual semantic feature information.

2.3. Improved Transformer Encoding Module

Based on the work of Xie [17], this paper proposes an enhanced Transformer model as an encoder for better establishing the correlation between Thangka images and text. The Thangka images and text exhibit strong correlation and closely related entity mappings. To address this, a modified Transformer model is constructed, incorporating a fusion of bidirectional residual connections and a masked attention mechanism.

2.3.1. Improved Residual Structure

The traditional Transformer model typically uses residual structures for layer normalization, as shown in Figure 3a with post-layer normalization (Post-LN) or in Figure 3b with pre-layer normalization (Pre-LN). Two commonly used variants are the Post-LN and Pre-LN transformers, which apply layer normalization after the output of each residual block or before the input of each residual block, respectively. While both variants have their advantages, they also have serious limitations: Post-LN leads to the gradient vanishing problem, hindering the training of deep Transformers, while Pre-LN leads to the representation collapse problem, limiting the model's capacity. In this paper, we introduce the Bidirectional Residual Connection [18], a novel Transformer architecture with Pre-Post-LN (PPLN) that combines the connections from Post-LN and Pre-LN, inheriting their advantages while avoiding their limitations. The residual is illustrated in Figure 3c. The vertical lines on the left and right sides represent two residual connections, where the left side is similar to the post-layer normalization module.

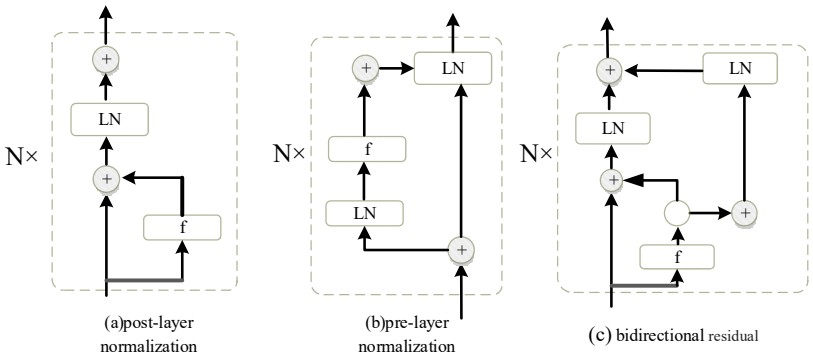

Figure 3. Residual structure of each type.

Assuming a Transformer network with N^* residual blocks, where the input matrix has a shape of N^*d (N represents the sequence length, and d represents the dimension of the sequence), let k denotes the tensor of the k-th block, X_k^{ln} represents the normalized tensor data, f_k represents the feed-forward neural network (in this paper, self-attention is chosen as the f_k), X_k^f represents the output tensor of the k-th block after the f_k function, and X_k^a represents the tensor obtained by adding X_k^f and X_k^{ln}. By normalizing X_k^a, the left-side output matrix X_{k+1}^{ln} can be obtained as shown in Equation (4):

$$X_k^a = X_k^{ln} + X_k^f = X_k^{ln} + f_k(X_k^a; wk); X_{k+1}^{ln} = \mathrm{LN}(X_k^a) \tag{4}$$

Similarly, for the right-side module, which is similar to the pre-layer normalization module, allowing gradients to flow directly to each block, X_k^d represents the input matrix from the previous layer and X_k^f represents the output value obtained after the feed-forward neural network calculation. Adding the two matrices gives the output matrix X_{k+1}^d on the right side, as shown in Equation (5):

$$X_{k+1}^d = X_k^d + X_k^f \tag{5}$$

Finally, the residual output y is computed by adding the representations from both residual structures, as shown in Equation (6):

$$\boldsymbol{y} = \boldsymbol{X_{N+1}^{ln}} + \boldsymbol{LN}\left(\boldsymbol{X_{N+1}^{d}} + \mathbf{1}\right) \tag{6}$$

2.3.2. Improved Masked Attention Mechanism

In the original Transformer model, the self-attention mechanism calculates the attention weights equally for all elements within the input vectors. While this method effectively captures the relationships between input vectors, it can lose the structural information present in input vectors with specific structures. However, in this paper, the established multi-granularity feature vectors for images and texts heavily rely on the structural information between entities. Therefore, the self-attention mechanism is improved in this paper by introducing a masking matrix M to the bidirectional residual Transformer network, aiming to preserve the structural information between entities. The modified structure is illustrated in Figure 4.

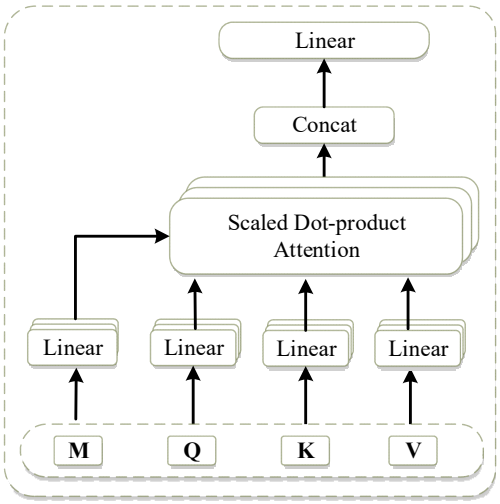

Figure 4. Masking attention mechanism.

The masking matrix is used to attenuate the influence of global semantic information on the structural information between entities in the multimodal feature vectors. This ensures the integrity of the structural information between entities. The masking matrix M is a diagonal matrix with dimensions $|\boldsymbol{X}| \times |\boldsymbol{X}|$, where the default values of the matrix elements are set to 1. However, to treat the attention between image region nodes and global sentence nodes as unrelated, the elements of the masking matrix M are set to $-\infty$. This reduces the attention between global sentence nodes and image region nodes, thereby preserving the structural information between entities. Similarly, to treat the attention between text region nodes and global image nodes as unrelated, the elements of the masking matrix M are also set to $-\infty$. This reduces the attention between text region

nodes and global image nodes, ensuring the integrity of the structural information between entities. The improved calculation process is shown in Equation (7):

$$atten(Q, K, V, M) = softmax\left(M\frac{Q^T K}{\sqrt{dk}}\right)V \tag{7}$$

Here, Q, K, and V are parameter matrices from the original attention calculation function, and M acts as the attention influence factor, implementing the masking for the attention in the two mentioned scenarios and reducing the influence of unrelated nodes on the final result.

With the improved Transformer, a sequence containing structural features between entities can be obtained, as shown in Equation (8):

$$Z = \left[x_u^I, X_u^I, x_v^T, X_v^T\right] \tag{8}$$

Here, x_u^I represents the local feature representation of the image, X_u^I represents the global feature representation of the image, x_v^T represents the local feature representation of the text, and x_v^T represents the global feature representation of the text.

2.4. Multigranularity Alignment Module

With the above processing, we obtain a series of multi-granular features for images and text, which next need to be modal aligned.

The main job of the alignment module is to map the input samples into a shared embedding space so that the similarity between them can be compared; the module performs this by learning a mapping function that converts the input samples into a low-dimensional vector representation. This has the advantage that it allows different types of inputs (e.g., text, images, or other data) to be represented uniformly as vectors, thus facilitating the computation of similarity between them. In matching problems, the calculation of similarity between positive and negative samples is critical. Positive samples are pairs of samples that belong to the same category or have similar attributes, while negative samples are pairs of samples that belong to different categories or have dissimilar attributes. By calculating the similarity between positive and negative samples, we can quantify the degree of difference between them. The purpose of this is to make the similarity between the positive samples as high as possible and the similarity between the negative samples as low as possible during the training process. In this way, we can train the model to learn the ability to distinguish between positive and negative samples, leading to better matching performance.

Based on the positive and negative samples' feature information, we construct four types of matching relationships: matching between the global region of the image and the text sentence, matching between the target region of the image and the text sentence, matching between the text entity and the global region of the image, and matching between the target region of the image and the text entity.

We calculate different matching scores and ensure that the matching score of the final positive image–text pair is greater than the matching score of the negative image–text pair. The calculation formulas are shown in Equations (9)–(11):

$$L_{g1} = T\left(X_i^I, X_i^T, X_j^T\right) + T\left(X_i^T, X_i^I, X_j^I\right) \tag{9}$$

$$L_{l1} = T\left(X_i^I, x_i^T, x_j^T\right) + T\left(X_i^T, x_i^I, x_j^I\right) \tag{10}$$

$$L_{l2} = T\left(x_i^I, x_i^T, x_j^T\right) + T\left(x_i^T, x_i^I, x_j^I\right) \tag{11}$$

In these equations, X_i^I, X_i^T, X_j^T represent the image output sample, text positive sample, and text negative sample in the positive image–text pair, respectively. The subscripts

i and ***j*** represent the data in the samples, and I and T correspond to the image and text modalities, respectively. L_{g1} is used to calculate the matching relationship in (1), L_{l1} is used to calculate the matching relationships in (2) and (3), and L_{l2} is used to calculate the matching relationship in (4). The function T represents the triplet loss function $T(u, P, N)$ [16], and its formula is shown in Equation (12):

$$T(u, P, N) = \max\left(\frac{1}{P}\sum_{p\in P} Sim(u, p) + \frac{1}{N}\sum_{n\in N} Sim(u, n) + d, 0\right) \quad (12)$$

In this paper, a set of image–text pairs (I_i, E_i) is established, which includes negative samples of text and negative samples of images. I_i represents the positive image sample, and E_i represents the positive text sample. In the loss function $T(u, P, N)$, u represents the training prediction sample, P represents the positive samples, and $\boldsymbol{N}$ represents the negative samples. The scalar d is used to control the distance range between the output sample $\boldsymbol{u}$ and the positive and negative samples P and N. Sim is the similarity function that calculates the similarity between the output sample and the positive/negative samples. In this paper, the cosine distance similarity function is used. The max() function is used to set the result to zero whenever the computed value of the function is less than or equal to zero. If it is greater than zero, the loss is a positive value, aiming to minimize the loss function to zero or less. The purpose of the loss function is to drive the output sample u to be as close as possible to the data in the positive samples $p \in P$, while keeping the output sample u as far away as possible from the data in the negative samples $n \in N$.

Based on the calculations of these two types of losses, we obtain the final total loss function as shown in Equation (13). The hyperparameters λ_0, λ_1, λ_2 are used to balance these losses, and their values are between 0 and 1.

$$L = \lambda_0 L_{g1} + \lambda_1 L_{l1} + \lambda_2 L_{l2} \quad (13)$$

The final total loss function L for the Thangka image–text domain is obtained by splitting and fusing the global and local semantic feature information between the Thangka image and text. This loss function encompasses the semantic relationship between the Thangka image regions and text phrases, and extracts the overall structural information from both modalities, thereby fully capturing the correlation between the Thangka image and text modalities.

3. Results

3.1. Experimental Data and Preprocessing

To conduct comparative and ablation experiments, this paper chose two datasets: a self-constructed Thangka image–text dataset and the widely employed Flickr30k dataset in the field of image–text matching [19].

The Thangka image–text dataset comprises 3000 samples of Thangka-themed images. In constructing image–text pairs for Thangka, we referenced the text structure of the commonly used Wikipedia dataset [20], combining it with the characteristics of Thangka themes, such as academic nomenclature, production techniques, thematic symbolism, cultural backgrounds, and more. For each Thangka theme category, we selected corresponding textual descriptions, resulting in an initial set of 2231 Thangka image–text pairs. Subsequently, data augmentation techniques were applied to expand the image and text datasets. For image data, in addition to conventional methods like brightness, contrast, rotation, and flipping, we employed a cut-and-replace approach to substitute entity parts within the Thangka images, thereby increasing the dataset size. To address the issue of limited textual data, we utilized Easy Data Augmentation (EDA) [21] for data augmentation. Following data augmentation, a total of 5500 Thangka image–text pairs were obtained. Consistent with the practice of the commonly used Wikipedia dataset for cross-modal matching, the constructed Thangka-themed image–text pairs were divided into a dataset in a 6:2:2 ratio. Specifically, 3300 pairs were randomly chosen for the training set, 1100 pairs for the valida-

tion set, and 1100 pairs for the testing set. An example of the Thangka image–text dataset is illustrated in Figure 5.

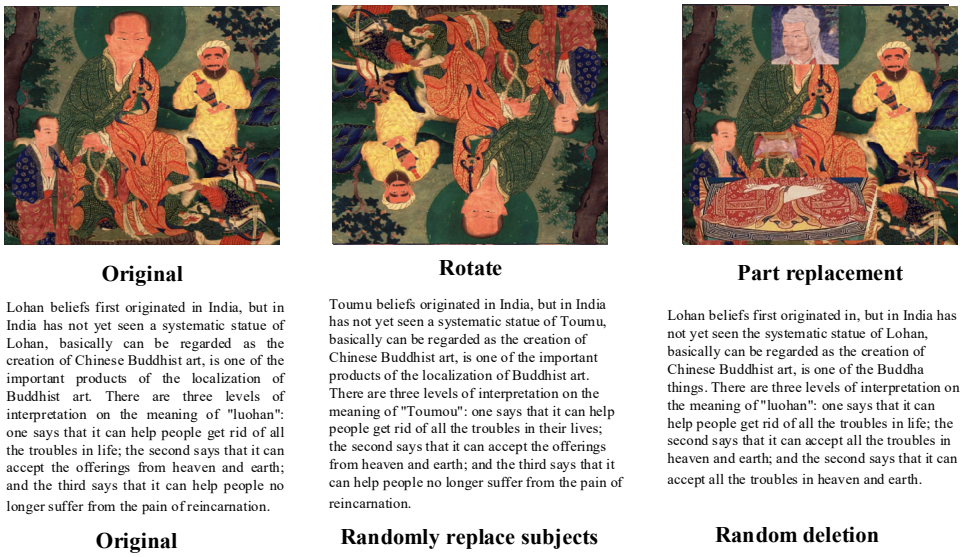

Figure 5. Example of a Thangka dataset.

3.2. Experimental Environment and Parameter Settings

The experiments were conducted on the following platform and hardware configuration: Ubuntu 16.04 operating system in the Internet open source and RTX A6000 GPU from NVIDIA, Santa Clara, CA, USA The experiments were implemented using Python 3.6.12 programming language, and the deep learning framework used was PyTorch 1.11.0.

The feature extraction module consisted of two sub-networks: one for the image modality and one for the text modality. For the image modality, the Thangka images were trained using a resolution of 384 × 384 and input into a pre-trained Faster R-CNN network to implement top-down attention. The global and local feature information of the images was obtained through training and an adaptive pooling layer. For the text modality, a pre-trained Bert-Bilstm-Crf model was used to train and extract entity and relationship information from the text. The Word2Vec word embedding model was then employed to obtain the word vector sequences of the text, with each word vector having a dimension of 300. The text was further processed by introducing a bidirectional GRU model with an adaptive pooling layer to capture contextual semantic feature information, resulting in the extraction of global and local feature information for the text. The bidirectional GRU model has a num_layers set to 2 and hidden_size set to 1024. The adaptive pooling layer consists of two layers: a linear layer and a fusion layer with fusion weights set to 0.7 for tokens and 0.3 for embeddings. In the Multigranularity Alignment Module, we initialize three fixed values (λ_0, λ_1, λ_2), all equal to 0.3, to start the training.

For the coding and alignment layers, the structure consisted of an encoding layer and an alignment layer. The hidden layer dimension was set to 1024, the attention heads were set to 16, and the internal dimension of the feed-forward network was set to 1024. These settings were used to learn multi-granularity semantic correlations between the image and text modalities.

During the training phase, an Adam optimizer with a batch size of 128 was employed. The learning rate was set to 0.0001, and the model was trained for 100 epochs. By configuring the experimental environment and parameters as described above, the researchers implemented and trained the model to conduct the experiments and obtain results for analysis.

3.3. Evaluation Metrics

To verify the accuracy of the experimental results, we designed two tasks for the Thangka image–text dataset: Thangka image matching text and Thangka text matching image, and conducted evaluations.

In this study, recall (R) [22] and OIPS were used as the evaluation metric for cross-modal matching between Thangka images and text. Recall refers to the ratio of matched samples returned by the matching model to the total number of matching samples in the dataset. It is used to calculate the probability of correctly matching positive samples. The calculation is shown in Equation (14):

$$R = \frac{a}{a+b} \times 100\% \tag{14}$$

Here, a presents the number of correctly matched samples in the positive samples, and b represents the number of incorrectly matched samples in the positive samples.

In the image–text matching task, recall is often set as $R@K$, which measures the percentage of test images or test sentences that find at least one correct result among the top K matching results. In this study, $R@1$, $R@5$, $R@10$ were used to represent the recall at the top 1, 5, and 10 results, respectively.

OIPS is the number of operational projects per second, which serves as a measure of a model's execution efficiency and responsiveness, with larger numbers indicating better model efficiency.

3.4. Model Comparison Experiment

We compared the proposed model with several powerful image–text matching baseline models. We divided the baseline models into two groups, with the first five models being VSE models and the remaining four models being pre-trained models. We chose the Thangka dataset and the Flickr30k public dataset as the evaluation datasets. For the models compared in this paper, on the Thangka dataset, we followed Section 3.2 and set the learning rate as 0.0001, epochs as 100, and batch size as 128 for training to ensure the fairness of the test results. On the public dataset, we tested the pre-trained models provided by the respective baseline model's official sources to validate the performance of our model on the public dataset. All experimental models underwent at least two to three tests, with results filtered to exclude deviations from the official data exceeding 5%, and the experimental results were averaged. The experimental results are shown in Tables 1 and 2.

- SCAN (Lee et al., 2018) [2]: Stacked Cross Attention for Image–Text Matching.
- CAMP (Wang et al., 2019) [23]: Cross-Modal Adaptive Message Passing for Text–Image Retrieval
- CAAN (Zhang et al., 2020) [3]: Context-Aware Attention Network for Image–Text Retrieval
- IMRAM (Chen et al., 2020) [4]: Iterative Matching with Recurrent Attention Memory for Cross-Modal Image–Text Retrieval
- GSMN (Liu et al., 2020) [24]: Graph Structured Network for Image–Text Matching
- ViLBERT (Lu et al., 2019) [6]: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks
- OSCAR (Li et al., 2020) [7]: Object-Semantics Aligned Pre-training for VisionLaguage Tasks
- CLIP (Radford et al., 2021) [25]: Learning Transferable Visual Models From Natural Language Supervision
- MVP (Li et al., 2022) [26]: Multi-stage vision-language pre-training via multi-level semantic alignment

Table 1. Comparison of the results of different methods on the Thangka dataset.

Mark	Model	Image-Text			Text-Image			OIPS
		R@1	R@5	R@10	R@1	R@5	R@10	
VSE	SCAN	23.2	53.5	72.1	16.3	55.6	68.9	48.7
VSE	CAMP	36.1	68.2	82.7	21.2	59.1	79.7	46.5
VSE	CAAN	37.5.	69.6	85.4	22.6	60.2	78.5	47.5
VSE	GSMN	41.8	72.6	88.3	23.7	63.1	79.7	42.5
VSE	IMRAM	42.9	71.2	87.6	25.7	62.5	79.1	44.8
Pre-trained	ViLBERT	39.0	70.5	84.1	23.5	62.7	78.8	16.1
Pre-trained	OSCAR	45.8	78.5	92.8	27.5	63.8	84.2	4.3
Pre-trained	CLIP	48.5	82.8	93.1	29.8	64.1	86.8	6.7
Pre-trained	MVP	49.8	84.4	93.8	30.4	66.3	85.5	5.2
VSE	**Our Method**	**54.7**	**85.4**	**94.3**	**33.6**	**70.3**	**86.4**	**32.7**

Table 2. Comparison of results of different methods on public datasets.

Mark	Model	Image-Text			Text-Image			OIPS
		R@1	R@5	R@10	R@1	R@5	R@10	
VSE	SCAN	61.8	87.5	93.7	45.8	74.4	83.0	48.3
VSE	CAMP	68.1	89.7	95.2	51.5	77.1	85.3	45.9
VSE	CAAN	70.1	91.6	97.2	52.8	79.0	87.9	48.1
VSE	GSMN	72.6	93.4	96.8	53.9	80.2	87.1	43.5
VSE	IMRAM	74.1	93.0	96.6	53.9	79.4	87.2	45.8
Pre-trained	ViLBERT	71.5	91.0	96.9	52.9	78.5	86.9	28.6
Pre-trained	OSCAR	76.4	94.3	98.2	59.4	84.4	91.6	5.4
Pre-trained	CLIP	78.2	95.6	98.2	67.0	87.7	93.4	3.1
Pre-trained	MVP	**78.9**	95.8	97.9	**69.2**	**88.1**	**94.2**	2.3
VSE	**Our Method**	77.9	**95.9**	**98.3**	57.3	85.7	92.3	**40.5**

Table 1 lists the recall rates (*R*@1, *R*@5, *R*@10) of different methods on the two matching tasks Image-to-Text and Text-to-Image of the Thangka image and text dataset. As can be seen from the table, without using any large-scale corpus for pre-training, the effect of this method on Thangka image and text data exceeds other models, and the recall rate *R*@1 on I2T and T2I is 54.7% and 33.6%, respectively, which are 9.5% and 10.6%, respectively, higher than the highest performance of the comparison method on the two matching tasks. It reflects that this method has better applicability and effect on the Thangka image and text dataset. At the same time, compared with the pre-trained model, the OIPS of this method is much better than other pre-trained models, which reflects that the pre-trained models of this method are much faster in training efficiency and execution speed.

For the I2T and T2I tasks on the Flickr30k dataset, the comparative results are shown in Table 2.

The experimental results of different methods on the public dataset Flickr30k are shown in Table 2. From the table, it can be observed that our method achieves a recall rate (R@1) of 77.9% for image-to-text (I2T) retrieval and 57.3% for text-to-image (T2I) retrieval on this dataset. Compared to the highest-performing IMRAM model in the best VSE method, our method improves the performance by 5.1% for I2T and 6.3% for T2I. Compared to the state-of-the-art pre-trained model MVP, our method has a lower R@1 recall rate for T2I, with a difference of approximately 2–3 percentage points for R@5 and R@10. For I2T, the R@1 recall rate is 3 percentage lower, while R@5 and R@10 are roughly the same or slightly higher.

It is noteworthy that our proposed method surpasses both pre-trained models, even without utilizing any large-scale corpora for pre-training. This approach has led to significant improvements in performance. Furthermore, our model outperforms the pre-trained

models in terms of OIPS, indicating that it is more efficient and faster in execution. By employing our method, we can precompute and cache visual and textual features, which only requires similarity calculations and sorting during the retrieval process. In conclusion, these results demonstrate that our model achieves a favorable balance between performance and efficiency.

3.5. Ablation Experiment

In order to verify the rationality of the design of the adaptive pooling layer and the improved Transformer in the model of this paper, ablation experiments were conducted. The results of the ablation experiments are listed in Table 3. Where A represents the adaptive pooling layer module, and T represents the improved Transformer module. NA indicates that the adaptive pooling layer module is not included, NT indicates that the improved Transformer module is not used, and Ours (A + T) indicates the method of this article.

Table 3. Ablation experiments on the Thangka dataset.

Model	Image-Text			Text-Image		
	R@1	R@5	R@10	R@1	R@5	R@10
NA + NT	42.1	65.2	84.1	22.3	50.1	70.9
A + NT	46.3	67.2	86.2	25.1	58.3	72.5
NA + T	49.6	74.4	87.7	29.6	67.2	74.3
Ours (A + T)	**54.7**	**85.4**	**94.3**	**33.6**	**70.3**	**86.4**

Based on the results in Table 3, it can be observed that by introducing the adaptive pooling layer module, the model is better able to extract semantic information from the Thangka text. This suggests that this method can more accurately capture entities and relationships in the text, thereby enhancing the effectiveness of image-text matching. Additionally, the introduction of the improved Transformer module also has a positive impact. This model combines bidirectional residual connections and masked attention mechanisms, enabling the effective extraction of global and local correlations between Thangka images and text. This indicates that the method can better align the semantic relationships between the images and text, further enhancing the effectiveness of image-text matching.

In summary, the introduction of the adaptive pooling layer and improved Transformer module allows for better extraction of semantic information from Thangka text and more accurate alignment of global and local correlations between Thangka images and text. As a result, the effectiveness of image-text matching is significantly improved.

4. Case Study

In order to better understand the image and text matching method proposed, a case study was conducted. The study provides two examples: image-to-text matching in Figure 6 and text-to-image matching in Figure 7. Examining Figure 6, it becomes evident that the image pertains to Lohan, and our model excels in accurately pairing it with the corresponding sentence while maintaining the correct order. The first text conveys, “Lohan beliefs initially originated in India; however, a systematic statue of Lohan has not yet been observed in India. Essentially, it can be regarded as a creation within Chinese Buddhist art and stands as a significant outcome of the localization of Buddhist art...” This text is accurate and aligns with the content of the image. At the same time, we notice that the text in the third position, although sharing the same sentence structure as the first one, is fundamentally incorrect. This indicates that the model excels at distinguishing cases where sentence segments are similar but the topics are different. This is mainly attributed to the model’s text enhancement. On the other hand, through Figure 7, we can see that our model matches some similar images based on the text, which means “Vajrasattva” is a transliteration of the Indian Sanskrit word, also translated as “Vajrapani”, “Vajrapani” and “Pushyen”, meaning “brave and powerful”, and the first picture in the

order is Vajrasattva, which corresponds to the text, indicating that our model's matching performance in matching images based on text is very good.

Figure 6. Top-5 image match on Thangka datasets. The ground-truth results are marked with red, and the wrong results are indicated in green.

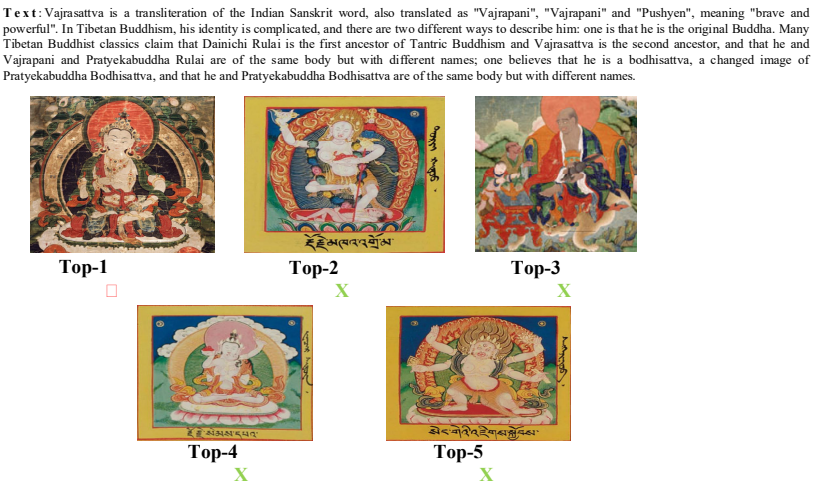

Figure 7. Top-5 text match on Thangka datasets. The ground-truth results are marked with red, and the wrong results are indicated in green.

5. Conclusions

In this paper, we propose a Thangka image-text matching method that combines an adaptive pooling layer and an improved Transformer. For the former, we build a Thangka image-text feature extraction model, where named entity recognition and the Faster-RCNN model are used to extract features for the text and image modalities, respectively. We fully extract modality-specific feature information and introduce an adaptive pooling layer that combines the features from token and embedding dimensions. This allows the visual and textual encoders to learn the best way to combine their features and generate fixed-sized embeddings. For the latter, we propose a masked attention mechanism and a Bidirectional Residual model to improve the traditional Transformer, enhancing its focus on local feature information in the Thangka domain. We evaluated the model on both public datasets and Thangka datasets, comparing it with traditional VSE baseline models and large-scale pre-trained models. The results showed that the adaptive pooling layer improved feature extraction capabilities, and the improved Transformer outperformed the traditional Transformer model in matching similarity calculation. Our proposed model achieves good performance while maintaining good computational efficiency.

However, there is still room for further improvement. For example, the traditional Faster-RCNN model performs poorly in processing complex Thangka images, such as those with multiple figures, due to the high similarity between different characters and the similarity between some characters and the background color. Exploring the integration of more effective target detection models, such as YOLO, into the feature processing for image-text matching will be a future research direction. In addition, since the available graphic data related to Thangka is still incomplete, this paper adopts graphic data augmentation to expand the original data. However, this makes the model more sensitive to noise and outliers during the training process. To address these issues, our future plans include exploring new algorithms and methods to improve the accuracy and robustness of the model, and attempting to apply the model to a wider range of datasets and application scenarios. Furthermore, we will supplement and enrich the Thangka image and text dataset to further enhance the performance of the model.

Author Contributions: Conceptualization, K.W.; methodology, T.W.; software, K.W.; validation, K.W.; formal analysis, J.W.; investigation, K.X.; resources, K.W.; data curation, K.W.; writing—original draft preparation, K.W.; writing—review and editing, X.G.; supervision, T.W.; funding acquisition, T.W. and X.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the 2023 Basic Research Operating Expenses of Central Universities (No. 31920230175), National Natural Science Foundation of China (No. 62166035), Natural Science Foundation of Gansu Province (No. 21JR7RA163) and Northwest University for Nationalities' Talent Recruitment Scientific Research Project (No. xbmuyjrc2023007).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are partially available on request from the corresponding author. The data are not publicly available due to their current restricted access.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Hotelling, H. Relations between Two Sets of Variates. In *Breakthroughs in Statistics: Methodology and Distribution*; Springer: Berlin/Heidelberg, Germany, 1992; pp. 162–190.
2. Frome, A.; Corrado, G.S.; Shlens, J.; Bengio, S.; Dean, J.; Ranzato, M.A.; Mikolov, T. Devise: A Deep Visual-Semantic Embedding Model. In *Advances in Neural Information Processing Systems*; NeurIPS: New Orleans, LA, USA, 2013; Volume 26.
3. Lee, K.H.; Chen, X.; Hua, G.; Hu, H.; He, X. Stacked cross attention for image-text matching. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 201–216.
4. Zhang, Q.; Lei, Z.; Zhang, Z.; Hu, H.; He, X. Context-aware attention network for image-text retrieval. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 3536–3545.
5. Chen, H.; Ding, G.; Liu, X.; Lin, Z.; Liu, J.; Han, J. Imram: Iterative matching with recurrent attention memory for cross-modal image-text retrieval. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 12655–12663.
6. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-Training of Deep Bidirectional Transformers for Language Understanding. *arXiv* **2018**, arXiv:1810.04805.
7. Lu, J.; Batra, D.; Parikh, D.; Lee, S. Vilbert: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks. In *Advances in Neural Information Processing Systems*; NeurIPS: New Orleans, LA, USA, 2019; Volume 32.
8. Li, X.; Yin, X.; Li, C.; Zhang, P.; Hu, X.; Zhang, L.; Wang, L.; Hu, H.; Dong, L.; Wei, F.; et al. Oscar: Object-Semantics Aligned Pre-Training for Vision-Language Tasks. In *Proceedings of the Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020*; Proceedings, Part XXX 16; Springer International Publishing: Berlin/Heidelberg, Germany, 2020; pp. 121–137.
9. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all You Need. In *Advances in Neural Information Processing Systems*; NeurIPS: New Orleans, LA, USA, 2017; Volume 30.
10. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards Real-Time Object Detection with Region Proposal Networks. In *Advances in Neural Information Processing Systems*; NeurIPS: New Orleans, LA, USA, 2015; Volume 28.
11. Dey, R.; Salem, F.M. Gate-Variants of Gated Recurrent Unit (GRU) Neural Networks. In Proceedings of the 2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS), Boston, MA, USA, 6–9 August 2017; pp. 1597–1600.
12. Diao, H.; Zhang, Y.; Ma, L.; Lu, H. Similarity reasoning and filtration for image-text matching. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtual Event, 2–9 February 2021; Volume 35, pp. 1218–1226.

13. Zhang, Z.; Shu, C.; Xiao, Y.; Shen, Y.; Zhu, D.; Xiao, J.; Chen, Y.; Lau, J.H.; Zhang, Q.; Lu, Z. Improving Visual-Semantic Embedding with Adaptive Pooling and Optimization Objective. *arXiv* **2022**, arXiv:2210.02206.
14. Xiaoran, G.; Ping, L.; Weilan, W. Chinese named entity recognition based on Transformer encoder. *J. Jilin Univ. (Eng. Ed.)* **2021**, *51*, 989–995.
15. Dai, Z.; Wang, X.; Ni, P.; Li, Y.; Li, G.; Bai, X. Named entity recognition using BERT BiLSTM CRF for Chinese electronic health records. In Proceedings of the 2019 12th International Congress on Image and Signal Processing, Biomedical Engineering and Informatics (CISP-BMEI), Suzhou, China, 19–21 October 2019; pp. 1–5.
16. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient Estimation of Word Representations in Vector Space. *arXiv* **2013**, arXiv:1301.3781.
17. Xie, S.; Zhang, H.; Guo, J.; Tan, X.; Bian, J.; Awadalla, H.H.; Menezes, A.; Qin, T.; Yan, R. ResiDual: Transformer with Dual esidual Connections. *arXiv* **2023**, arXiv:2304.14802.
18. Schroff, F.; Kalenichenko, D.; Philbin, J. Facenet: A unified embedding for face recognition and clustering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 815–823.
19. Plummer, B.A.; Wang, L.; Cervantes, C.M.; Caicedo, J.C.; Hockenmaier, J.; Lazebnik, S. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 2641–2649.
20. Vrandečić, D.; Krötzsch, M. Wikidata: A free collaborative knowledgebase. *Commun. ACM* **2014**, *57*, 78–85. [CrossRef]
21. Wei, J.; Zou, K. Eda: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks. *arXiv* **2019**, arXiv:1901.11196.
22. Buckland, M.; Gey, F. The relationship between recall and precision. *J. Am. Soc. Inf. Sci.* **1994**, *45*, 12–19. [CrossRef]
23. Wang, Z.; Liu, X.; Li, H.; Sheng, L.; Yan, J.; Wang, X.; Shao, J. Camp: Cross-modal adaptive message passing for text-image retrieval. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 5764–5773.
24. Liu, C.; Mao, Z.; Zhang, T.; Xie, H.; Wang, B.; Zhang, Y. Graph structured network for image-text matching. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 10921–10930.
25. Radford, A.; Kim, J.W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. Learning transferable visual models from natural language supervision. In Proceedings of the 38th International Conference on Machine Learning, Virtual Event, 1 July 2021; pp. 8748–8763.
26. Li, Z.; Fan, Z.; Tou, H.; Wei, Z. Mvp: Multi-Stage Vision-Language Pre-Training via Multi-Level Semantic Alignment. *arXiv* **2022**, arXiv:2201.12596.