MBA: A Multimodal Bilinear Attention Model with Residual Connection for Abstractive Multimodal Summarization

Xia Ye¹, Zengying Yue¹* and Ruiheng Liu¹

¹ Academy of Combat Support, Rocket Force University of Engineering, Xi’an, Shanxi, 710025, China
*Corresponding author’s e-mail: zengying2020@gmail.com

Abstract. The combination of vision and natural language modalities has become an important topic in both computer vision and natural language processing research communities. Multimodal summarization has received unprecedented attention with the rapid growth of multimodal information. This paper proposes MBA which consists of pre-trained feature extractors, text encoder, image encoder, multimodal bilinear attention fusion module, and summary decoder to complete abstractive multimodal summarization task. A residual network is added to the model to enhance the textual modality information and alleviate the modality-bias problem. Experiments show that the model is better than the baseline models and performs better than text summarization methods that ignore visual modality.

1. Introduction

Current summarization technology mainly focuses on text modal data, however appointment of visual modal data can improve the quality of the generated summary. Multimodal features are characterized by small intra-modal variance and large inter-modal variance (that is, the difference in the same-modal features of different samples is small, but the difference in different modal features is large), and there is a certain correlation between the modal. Therefore, multi-modal summaries can improve the quality of generated summaries by using visual modal features[1]. Studies have shown[2] that multi-modal output summaries are easier to improve reader satisfaction than text summaries.

The current research of multimodal summarizations[3] are mostly based on multimodal representation learning, multi-modal feature fusion, model structure design and other steps. Learning the representation of input data is the core issue of deep learning. The rapid development of pre-training models in recent years has brought new ideas to multimodal representation learning. Currently, the most commonly used multimodal features fusion methods are concatenation[4, 5], element-wise product, element-wise sum[6], etc. These simple operations are not enough to model the complex relationship between modalities. The mainstream generative multimodal summarization model architecture is mostly based on RNN and attention mechanism, and is more inclined to text modal data training. There is no optimization for multimodal data, which will lead to modality-bias[7] (the system tends to only optimize the text summary generation process, while the image quality is ignored during training). Moreover, the conventional attention mechanism uses the dot product method to fuse multimodal representations. We believe that this is not enough to fully capture the complex association between two different modalities. In this paper, we propose a multimodal bilinear attention fusion module based on multimodal compact bilinear pooling (MCB)[8] to obtain a joint representation.
The closest work to ours is that of Zhu et al. [2], who first applied the pointer generation network to multimodal summarization with multimodal output (MSMO) tasks. To our best knowledge, we combine pointer generation network with MBA to complete MSMO task for the first time. The main contributions of our work are as follows:

- We propose the MBA model and apply it to MSMO tasks.
- We introduce the pre-trained model to extract multimodal features and apply them to MSMO to make the model more general.
- We introduce residual connection to the multimodal summary generation framework to avoid modality-bias.

2. Related Work

MSMO refers to input text and several related pictures, output the corresponding summary, and select one of the most important pictures from the input pictures, as shown in Figure 1.

![Figure 1. The example of MSMO.](image)

2.1. Representation Learning

Representation learning is the core issue of deep learning. It is particularly difficult for multimodal tasks to perform data representation learning across different modalities. Leveraging pre-trained representations with the desired properties, such as properties suitable for zero-shot or few-shot learning, is often an effective solution to this issue[9].

Through the last CNN layer of the image classification network, the image feature representation can be acquired. The well-known models such as VGGNet[10], and ResNet[11] have been fully verified, and they are widely used in various computer vision (CV) tasks. For convolution operations, the methods to improve the accuracy of the model are mainly focused on deepening or widening the network, spatially fusing more feature, or extracting multi-scale spatial information. ResNeXt[12] proposes the concept of cardinality (channel grouping). Through experiments, the author finds that increasing the number of channels is more effective than increasing the depth and width. SENet[13] proposes a brand new Squeeze-and-Excitation (SE) structure, which is essentially an attention mechanism for channels. It controls the size of the important features between the channels are enhanced, and the unimportant features are weakened, so that the extracted features are more directional. Additionally, visual representations with more relationships to semantics can be acquired from object detection models, such as region-based CNN [14] (R-CNN), and Faster R-CNN [15].

Language representations has experienced word embedding and context embedding stages. The Neural Network Language Model [16] (NNLM), which estimates the probability of a text sequence by factorizing the sequence into word probabilities using a chain rule for probability, realizes the mapping from symbol space to vector space, marking the birth of word embedding. Word2vec[17] and GloVe[18] further improve the word embedding algorithm and enhance the semantic features of word embedding. ELMO[19] integrates contextual features into vector space for the first time through an
autoregressive language model composed of two-layer bidirectional LSTM. The GPT series[20, 21] uses Transformer encoder to form an autoregressive language model, which has stronger language generation capabilities and can learn long-distance dependence. Bidirectional Encoder Representations from Transformers (BERT)[22] reconstructs the original data from the input sequence with noise, can encode context information well, and has stronger natural language understanding (NLU) capabilities. In addition to word and sub-word levels, we can also learn language representations at phrase, sentence, and paragraph levels.

2.2. Multimodal Features Fusion
Multimodal features fusion is a key research topic in multimodal research, which integrates information extracted from unimodal data sources into a stable multimodal representation[9]. Fusion methods can be divided into early fusion (based on features), late fusion (based on decision-making) and intermediate fusion based on the stage in which fusion occurs during the related process[23]. Because early fusion and late fusion will inhibit intra-modal and inter-modal interaction and may cause vector redundancy, intermediate fusion is the more effective method. Anastasopoulos et al. propose a method of extracting unimodal high-level features using pre-training models and then fusing them through concatenation[4]. The model combines the outputs of attention mechanisms over each modal sequence by two approaches, flat and hierarchical. Besides simple operation-based and attention based fusion methods, bilinear pooling[24] is also a method commonly used to fuse visual feature vectors with textual feature vectors to create a joint representation space by computing their outputs. Because bilinear pooling can cause explosive growth in dimensionality, low-dimensional approximations can be used to obtain compact bilinear representations. MCB[8] with the count sketch projection function and multimodal low-rank bilinear pooling (MLB)[25] using Hadamard product can efficiently and expressively combine multimodal features. Kim et al.[26] propose BAN to model the bilinear interaction of bimodal input, while use MLB to extract the joint representations of the bimodality.

2.3. Multimodal Summarization
In the field of text summarization, the introduction of multi-modal information has also been practiced, and the main consideration is how to represent and fuse multimodal information. An attentional hierarchical Encoder-Decoder model[3] with a multimodal attentional mechanism is proposed to summarize a text document and its accompanying images simultaneously. Zhu et al. use the pointer generation network in the task of MSMO, and propose a multimodal features fusion method with a hierarchical attention mechanism[2]. To alleviate modality-bias, a multimodal objective function[7] with the guidance of multimodal reference is proposed to use the loss from the summary generation and the image selection.

3. Our Model and Methods
As shown in figure 2, our proposed model consists of four modules: text encoder, image encoder, multimodal bilinear attention fusion module, and summary decoder.

3.1. Pointer-Generator Network
The pointer-generator network is proposed to solve the problem of OOV and repetition in abstractive text summarization. Zhu et al. extend the pointer-generation network to MSMO tasks[2]. Their model consists of a bidirectional LSTM (text encoder), VGG19 which pre-trained on ImageNet dataset (image encoder), a multimodal attention model, and a unidirectional LSTM (decoder). The text encoder completes the mapping of articles to hidden states \( h_t \). Then the context vector \( c_{ext} \) is computed by the attention mechanism as follows:

\[
\begin{align*}
    e'_i &= v^T \tanh(W_f h_t + W_s s_i + W_c c_{cov'}) \\
    \alpha' &= \text{softmax}(e')
\end{align*}
\]
\[ c'_{ct} = \sum_{i} \alpha'_{i} h_i \]  

where \( s_t \) denotes the decoder state at the current timestep, \( \text{cov}' \) denotes the coverage vector[27] which is the sum of attention distributions over all previous decoding timesteps (initialized to zero vector at the first timestep):

\[ \text{cov}' = \sum_{t=0}^{t-1} \alpha' \]  

The coverage vector aims to alleviate the problem of repetition. During training, the input \( x_t \) of the decoder at timestep \( t \) denotes the embedding of previous words of the reference summary. In addition, during predicting, the input \( x_t \) denotes the previous words emitted by the decoder. In order to solve the problem of unregistered words, they proposed the concept of pointer which is the generation probability \( [0,1] \). For timestep \( t \), the generation probability is calculated as Eq. 5. The model can generate a word with probability \( p_g \) from the vocabulary distribution \( p_v \), and copy a word from the input articles by sampling from the attention distribution \( \alpha' \).

\[
\begin{align*}
p_g &= \sigma(W_h c_t + W_s s_t + W_x x_t) \\
p_v &= \text{softmax}(V_c [s_s, c_s] + b_c + b) \\
p_w &= p_s p_v(w) + (1 - p_g) \sum_{w_i = w} \alpha'_i
\end{align*}
\]  

The vocabulary distribution \( p_v \) is calculated by concatenating the decoder state \( s_t \) and the context vector \( c_t \) and then feeding them into two linear layers. \( p_w \) represents the final probability distribution over the extended vocabulary, which denotes the union of the vocabulary and all words in the input articles. Finally, the loss consists of the negative log likelihood of the target word \( w'_t \) for timestep \( t \) and the coverage loss with a hyperparameter \( \lambda \):

\[ L_t = -\log p_{w'_t} + \lambda \sum_{t} \min(\alpha'_t, \text{cov}'_t) \]
3.2. MCB

Algorithm 1. Multimodal Compact Bilinear

1: input: \( c_1 \in R^{d_1}, c_2 \in R^{d_2} \)
2: output: \( \Phi(c_1, c_2) \in R^d \)
3: procedure MCB\( (c_1, c_2, d_1, d_2, d) \)
4: for \( k \) in \{1, 2\} do
5: if \( h_k, s_k \) not initialized then
6: for \( i \) in \{1, \ldots, n_k\} do
7: sample \( h_k[i] \) from \{1, \ldots, d\}
8: sample \( s_k[i] \) from \{-1,1\}
9: \( c'_k = \Psi(c_1, c_2, d_1, d_2) \)
10: \( \Phi = \text{FFT}^{-1}(\text{FFT}(c'_k) \square \text{FFT}(c'_k)) \)
11: return \( \Phi \)
12: procedure \( \Psi(c, h, s, d) \)
13: \( y = [0, \ldots, 0] \)
14: for \( i \) in \{1, \ldots, d\} do
15: \( y[h[i]] = y[h[i]] + s[i] \cdot c[i] \)
16: return \( y \)

We first review bilinear models\[24\] which take the outer product of two vectors \( c_1 \in R^{d_1} \) and \( c_2 \in R^{d_2} \) and learn a linear model \( W \).

\[
y = W[c_1 \otimes c_2]
\]

where \( \otimes \) denotes the outer product and \( [] \) denotes linearizing the matrix in a vector. When the dimension of \( y \) is \( d \), \( W \) would have \( d_1 \times d_2 \times d \) parameters, which leads to very high memory consumption and high computation times. In order to avoid calculating the outer product explicitly, Akira et al. proposed a method of expressing the outer product of two vectors using the convolution of two count sketches\[8\]. The Count Sketch projection function, which projects \( c_1 \in R^{d_1} \) and \( c_2 \in R^{d_2} \) to \( y \in R^d \), is formalized as Eq. 10.

\[
\Psi(c_1 \otimes c_2, h, s) = \Psi(c_1, h, s) \ast \Psi(c_2, h, s)
\]

where \( \ast \) denotes the convolution operator, \( h \) and \( s \) are the intermediate vectors of the projection function, which are initialized randomly from a uniform distribution and remain constant for future invocations of count sketch. In addition, as described by the convolution theorem, convolution in the time domain is equivalent to element-wise product in the frequency domain. Therefore, Eq. 10 can be described as Algorithm 1.
3.3. Text Encoder

The text encoder consists of a pre-trained BERT and bidirectional LSTM. WordPiece algorithm of BERT can effectively alleviate the impact of out-of-word (OOV) and improve the effect of the model. For sequential tokens \((x_0, x_1, \ldots, x_n)\), the pre-trained BERT maps \(x_i\) into distributed representations rich in syntactic and semantic features \(w_i \in \mathbb{R}^{768}\). For \(x_i\), its input representation is constructed by summing the corresponding token, segment, and position embeddings. Then according to the principle of transfer learning, the extracted token-level representation is used to input the bidirectional LSTM to avoid learning from scratch. Bidirectional LSTM is responsible for the encoding of articles and completes the mapping of articles to a sequence of encoder hidden states \(h_i \in \mathbb{R}^{2048}\). Then we can obtain the text context vector \(c_{txt} \in \mathbb{R}^{1024}\) through Eq. 1 to 3.

3.4. Image Encoder

In the field of visual question answering (VQA), ResNet\([11]\) has become a commonly used visual feature extraction model, and in the field of multimodal summarization, VGGNet is still used as a visual feature extraction model. In this paper, we use the SE-ResNeXt50 model\([12]\) pre-trained on the ImageNet dataset as the image encoder to extract the 2048-dimensional feature of the global average pool layer as vector \(g_i \in \mathbb{R}^{2048}\). In the MSMO dataset, each sample contains more than one image. In order to improve the quality of image representation, we apply a clustering algorithm to sample \(M\) high-quality images in the image set. The relevant details are discussed in Section 4.1. We concatenate \(M\) vectors, which are the representations of images, into a matrix \(G = (g_1, \ldots, g_M)\). In Chapter 4, we analyze the effects of the two models by an ablation study.

3.5. MBA Module

Our MBA Module consists of a textual attention layer, a visual attention layer, and a multimodal bilinear attention layer, as shown in Figure 3. To facilitate the description, we simplify the MCB model to the following equation:

\[
y = \text{MCB}(v_1, v_2)
\]  

(11)

The attention mechanism can focus on different positions of the input sequence and has been widely used in the field of computer vision and natural language processing. We combine the attention mechanism with MCB and apply it to multimodal information fusion. As we discussed in section 3.1, we can obtain a text context vector \(c_{txt} \in \mathbb{R}^{1024}\) through the textual attention layer by the Eq. 1-3. Then we obtain a visual context vector \(c_{imag} \in \mathbb{R}^{1024}\) through the visual attention layer as follows:

\[
e' = v^T \tanh(W' g_i + U s_i + \text{cov}_{imag}')
\]  

(12)

\[
\alpha' = \text{softmax}(e')
\]  

(13)

\[
c'_{imag} = \sum_i \alpha'_i g_i
\]  

(14)

where \(\text{cov}_{imag}'\) denotes the visual coverage vector, which is initialized to zero vector in the beginning, and is calculated as Eq. 4. The visual coverage vector can not only reduce the repeated attention to multi-modal information, but also can be used as an image saliency measure. We can decide which image is most relevant to the text summary as the image summary through the accumulated visual coverage vector \(\text{cov}_{imag}'\) at the last time step of the decoder\([2]\), where \(t'\) denotes the last time step and \(j\) denotes the \(j\)-th image of the sample. To fuse the text and visual context information, we proposed a multimodal bilinear attention layer. In addition, the multimodal bilinear attention is described as follows:
\[
e_{\text{txt}} = v_{\text{txt}}^T (W_{\text{txt}} e_{\text{txt}} + U_{\text{txt}} s_t) \\
e_{\text{imag}} = v_{\text{imag}}^T (W_{\text{imag}} e_{\text{imag}} + U_{\text{imag}} s_t) \\
\alpha_{\text{txt}} = \text{softmax}(e_{\text{txt}}) \\
\alpha_{\text{imag}} = \text{softmax}(e_{\text{imag}}) \\
e_{\text{mm}} = \text{MCB}(\alpha_{\text{txt}} e_{\text{txt}}, \alpha_{\text{imag}} e_{\text{imag}})
\]

where \(e_{\text{mm}}\) is the \(e_i\) in Eq. 5 and 6. We believe that the textual modality is more important in the MSMO task, so we apply a residual connection in our model to enhance the textual modality information and alleviate modality-bias problem, as is shown in Figure 2. Finally, to alleviate repetition, we introduce the text coverage loss and visual coverage loss into the final loss function:

\[
L_t = -\log p_{c_t} + \lambda \sum_j \min(\alpha_{\text{txt}}, \text{cov}_j) + \gamma \sum_j \min(\alpha_{\text{imag}}, \text{cov}_{\text{imag},j})
\]

where \(\lambda\) and \(\gamma\) are hyperparameters, \(j\) denotes \(j\)-th image of the sample.

Figure 3. The structure of MBA module.

4. Experience
The hyperparameters in our model are similar to Zhu et al.[2], except that our model has 1024-dimensional hidden states and 768-dimensional word embeddings, and we set maximum number of images \(M\) to 7. In addition, when we set both \(\lambda\) and \(\gamma\) to 1, we can get better scores (we only test 0.5, 1, and 1.5).

4.1. Data Preparation
We use the MSMO dataset which consists of training dataset, validation dataset and test dataset, more details are illustrated in Table 1.

Table 1. Dataset statistics. Each article is paired with more than one images, and each image is paired with a caption. AvgTokens(A) and AvgTokens(S) denote the average number of tokens in articles and summaries respectively. In addition, AvgImages denotes the average number of images in articles.

|            | train  | valid | test  |
|------------|--------|-------|-------|
| Articles   | 293,965| 10,355| 10,261|
| AvgTokens(A)| 726    | 718   | 720   |
| AvgImages  | 6.37   | 7.12  | 6.85  |
| AvgTokens(S)| 69     | 71    | 70    |
We use the hash function and the URL of the crawled data set to achieve spatial alignment of text and image data. Through experiments, we find that there is a lot of noise in the dataset, and data cleaning can improve model performance. For text data, we clean the training corpus as close as possible to the pre-trained corpus to alleviate OOV problem. The specific operation is as follows:

- We delete the stop words that were removed during BERT pre-training.
- We fill in the missing period (a sign used in the model to identify whether it is a sentence).
- All numbers greater than 9 in the pre-training word embedding are replaced by the symbol "#". We simulate this preprocessing step to further clean the text data.

For image data, we first clean up images with too large or too small aspect ratios (mostly screenshots of web comments). To preserve the important information in the image as much as possible, we adapt the method of zooming after the centralized cropping to avoid the image scale imbalance. Finally and most importantly, we use the K-means algorithm to select the most important $M$ (which is described in Section 3.4) images instead of selecting them by the order of the position in the article (the algorithm adopted by Zhu et al.[2]).

### 4.2. Effectiveness of the Text Summarization

All models were trained for 50,000 iterations on 1 GPU (TITAN RTX 24GB), then we evaluate the best iteration on test dataset. To be fair, we set the max-encoder-sequence to 400. We adapt ROUGE[28], which is widely used, to evaluate effectiveness of the summarizations. We introduce four abstractive summarization methods to our evaluation system. Seq2seqatt[29] is a sequence-to-sequence RNN model with attention mechanism. PGN[27] denotes the pointer-generator network for abstract summarization. HNNatt[3] is the abstractive text-image summarization using multimodal attentional hierarchical RNN. MSMO[2] refers to our baseline which extends the pointer-generation network to multimodal summarization tasks. It’s worth explaining that we use the global feature model of MSMO. In addition, MBA is our model. The comparison with abstractive summarization methods is shown in Table 2.

#### Table 2. ROUGE scores on abstractive summarization methods.

| Method    | Rouge-1 | Rouge-2 | Rouge-L |
|-----------|---------|---------|---------|
| Seq2seqatt| 31.41   | 13.29   | 32.70   |
| PGN       | 39.61   | 18.01   | 36.41   |
| HNNatt    | 32.59   | 12.06   | 23.88   |
| MSMO      | 40.10   | 17.96   | 36.50   |
| MBA       | 40.43   | 18.27   | 36.96   |

Results in Table 2 show that our model performs better than others, and visual modality can improve the quality of text summarization. We can prove that the reason of the decrease in ROUGE scores which is discussed in paper[2] is right. Before we applied data cleaning, our model scored very low. Especially for images, more accurate image filtering algorithms can continue to improve model performance.

### 4.3. Effectiveness of the Image Summarization

We use image precision[2] (IP) to measure the salience of image of image summarization. The IP is calculated as follows:

$$IP = \frac{|\text{ref}_{imag} \cap \text{rec}_{imag}|}{|\text{rec}_{imag}|}$$

Where $\text{ref}_{imag}$, $\text{rec}_{imag}$ denote reference images and recommended images respectively. We evaluate the salience of image by IP on HNNatt, MSMO, and our model MBA. For HNNatt, we select the first image from ImgSum[3]. The result is shown in Table 3.
Table 3. Evaluation of image summarization.

| Model | IP  |
|-------|-----|
| HNNatt | 57.49 |
| MSMO  | 59.20 |
| MBA   | 61.87 |

HNNatt adopts an attention score method to select images that are more relevant to the summary. From Table 3, visual coverage method performs better than the attention score method. The reason for the low score of MSMO is that they did not filter the images in the dataset.

4.4. Ablation Studies

In this section, we perform ablation experiments over the MBA module and the residual connection in order to better understand their relative importance. The first row of Table 4 shows the impact of replacing the MBA layer with a standard attention mechanism on model performance. Then we removed the second MBA layer and the residual connection, and the first MBA layer directly output to the decoder. The result is shown in the second row of Table 4.

Table 4. Ablation over the MBA module and the residual connection.

| Model | Rouge-1 | Rouge-2 | Rouge-L |
|-------|---------|---------|---------|
| ATT   | 39.82   | 18.04   | 36.75   |
| MBA-Res | 38.79   | 17.19   | 35.83   |
| MBA   | 40.43   | 18.27   | 36.96   |

From Table 4, we can conclude that both the residual connection and the MBA layer play a vital role in the performance of the model, but the residual connection is more important. We guess that this is because the information of the text modality is more important to the model in the MSMO task.

5. Conclusion

In this paper, we focus on the multimodal information fusion. We propose the MBA module to better model the relationship between multimodal information. To make the model more versatile, we use pre-trained BERT to extract word embedding. In addition, we introduce the residual connection to the multimodal summarization framework.

Acknowledgments

The research work described in this paper has been supported by the National Natural Science Foundation of China (Grant No.62006240). We thank Tingting Liu for her careful reading of our paper and her many insightful comments and suggestions. We thank Kun Bai for the helpful discussions.

References

[1] Li H, Zhu J, Liu T, et al. Multi-modal Sentence Summarization with Modality Attention and Image Filtering[C]//IJCAI. 2018: 4152-4158.
[2] Zhu J, Li H, Liu T, et al. MSMO: Multimodal summarization with multimodal output[C]//Proceedings of the 2018 conference on empirical methods in natural language processing. 2018: 4154-4164.
[3] Chen J, Zhuge H. Abstractive text-image summarization using multi-modal attentional hierarchical rnn[C]//Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. 2018: 4046-4056.
[4] Anastasopoulos A, Kumar S, Liao H. Neural language modeling with visual features[J]. arXiv preprint arXiv:1903.02930, 2019.
[5] Wang H, Meghawat A, Morency L P, et al. Select-additive learning: Improving generalization in multimodal sentiment analysis[C]//2017 IEEE International Conference on Multimedia and Expo (ICME). IEEE, 2017: 949-954.
[6] Pérez-Rúa J M, Vielzeuf V, Pateux S, et al. Mfas: Multimodal fusion architecture search[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 6966-6975.

[7] Zhu J, Zhou Y, Zhang J, et al. Multimodal summarization with guidance of multimodal reference[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2020, 34(05): 9749-9756.

[8] Fukui A, Park D H, Yang D, et al. Multimodal compact bilinear pooling for visual question answering and visual grounding[J]. arXiv preprint arXiv:1606.01847, 2016.

[9] Zhang C, Yang Z, He X, et al. Multimodal intelligence: Representation learning, information fusion, and applications[J]. IEEE Journal of Selected Topics in Signal Processing, 2020, 14(3): 478-493.

[10] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[J]. arXiv preprint arXiv:1409.1556, 2014.

[11] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.

[12] Xie S, Girshick R, Dollár P, et al. Aggregated residual transformations for deep neural networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 1492-1500.

[13] Hu J, Shen L, Sun G. Squeeze-and-excitation networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 7132-7141.

[14] Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2014: 580-587.

[15] Ren S, He K, Girshick R, et al. Faster r-cnn: Towards real-time object detection with region proposal networks[J]. arXiv preprint arXiv:1506.01497, 2015.

[16] Bengio Y, Ducharme R, Vincent P, et al. A neural probabilistic language model[J]. The journal of machine learning research, 2003, 3: 1137-1155.

[17] Mikolov T, Chen K, Corrado G, et al. Efficient estimation of word representations in vector space[J]. arXiv preprint arXiv:1301.3781, 2013.

[18] Pennington J, Socher R, Manning C D. Glove: Global vectors for word representation[C]//Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). 2014: 1532-1543.

[19] Peters M E, Neumann M, Iyyer M, et al. Deep contextualized word representations[J]. arXiv preprint arXiv:1802.05365, 2018.

[20] Radford A, Narasimhan K, Salimans T, et al. Improving language understanding by generative pre-training[J]. 2018.

[21] Radford A, Wu J, Child R, et al. Language models are unsupervised multitask learners[J]. OpenAI blog, 2019, (1): 8.

[22] Devlin J, Chang M W, Lee K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.

[23] LeCun Y, Bengio Y, Hinton G. Deep learning[J]. nature, 2015, 521(7553): 436-444.

[24] Tenenbaum J B, Freeman W T. Separating style and content with bilinear models[J]. Neural computation, 2000, 12(6): 1247-1283.

[25] Kim J H, On K W, Lim W, et al. Hadamard product for low-rank bilinear pooling[J]. arXiv preprint arXiv:1610.04325, 2016.

[26] Kim J H, Jun J, Zhang B T. Bilinear attention networks[J]. arXiv preprint arXiv:1805.07932, 2018.

[27] See A, Liu P J, Manning C D. Get to the point: Summarization with pointer-generator networks[J]. arXiv preprint arXiv:1704.04368, 2017.

[28] Lin C Y. Rouge: A package for automatic evaluation of summaries[C]//Text summarization branches out. 2004: 74-81.
[29] Nallapati R, Zhou B, Gulcehre C, et al. Abstractive text summarization using sequence-to-sequence rnns and beyond[J]. arXiv preprint arXiv:1602.06023, 2016.