YNU-HPCC at SemEval-2022 Task 5: Multi-Modal and Multi-label Emotion Classification Based on LXMERT

Chao Han, Jin Wang and Xuejie Zhang
School of Information Science and Engineering
Yunnan University
Kunming, China
hc_super@mail.ynu.edu.cn, {wangjin, xjzhang}@ynu.edu.cn

Abstract

This paper describes our system used in the SemEval-2022 Task5 Multimedia Automatic Misogyny Identification (MAMI). This task is to use the provided text-image pairs to classify emotions. In this paper, We propose a multi-label emotion classification model based on pre-trained LXMERT. We use Faster-RCNN to extract visual representation and utilize LXMERT's cross-attention for multi-modal alignment. Then we use the Bilinear-interaction layer to fuse these features. Our experimental results surpass the F_1 score of baseline. For Sub-task A, our F_1 score is 0.662 and Sub-task B's F_1 score is 0.633. The code of this study is available on GitHub[1].

1 Introduction

In social networks, meme is mainly used to express the emotion of netizen. It usually consists of text and images. But at the same time, memes also convey some negative emotions, such as negative comments about women. SemEval-2022 Task5: Multimedia Automatic Misogyny Identification (MAMI) (Fersini et al., 2022) focuses on identifying whether meme conveys negative emotions towards women.

- Sub-task A: a basic task about misogynous meme identification, where a meme should be categorized either as misogynous or not misogynous;
- Sub-task B: an advanced task, where the type of misogyny should be recognized among potential overlapping categories such as stereotype, shaming, objectification, and violence.

Since the Transformer (Vaswani et al., 2017) and BERT (Devlin et al., 2019) models were proposed, researchers have begun to work on image and text multi-modality work in recent years, in addition to using one modality such as only image or text. Nowadays, for multimodal models, they can be divided into two categories, single-stream model and dual-stream model. In the single-stream model, language information and vision information are fused at the beginning and directly input into the encoder. Some representative single-stream models include ImageBERT (Qi et al., 2020), Unicoder VL (Li et al., 2020), VL-BERT (Su et al., 2020), VisualBERT (Li et al., 2019), etc. In the dual-stream model, in addition to the LXMERT, we will introduce below, there were ViLBert (Lu et al., 2019) and UNIMO (Li et al., 2021), etc.

As for emotion recognition, in previous tasks, there are also emotion classification tasks based on multi-modal graphics and text, such as Zhu et al. (2021) used text-CNN and ALBERT to Identify the persuasion skills of Meme. Peng et al. (2020) used the adversarial learning of sentiment word representations for sentiment analysis. A tree-structured regional CNN-LSTM (Wang et al., 2020) and dynamic routing in a tree-structured LSTM (Wang et al., 2019) were used for dimensional sentiment analysis. In previous SemEval competitions, Tian et al. (2021) extracted heterogeneous visual representations (i.e., face features, OCR features, and multimodal representations) and explored various multimodal fusion strategies to combine the textual and visual representations. In addition, in multimodal analysis combining images and text, Yuan et al. (2020) proposed a parallel channel ensemble model combining BERT embedding, BiLSTM, attention and CNN, and ResNet for sentiment analysis of memes.

The main difficulty of multi-modality is how to extract the two modalities' features and express the semantics more accurately, which involves the representation of multi-modality, the alignment between multi-modality, and the fusion of multi-

[1] https://github.com/HC-super/SemEval-2022-Task-5

Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022), pages 748 - 755

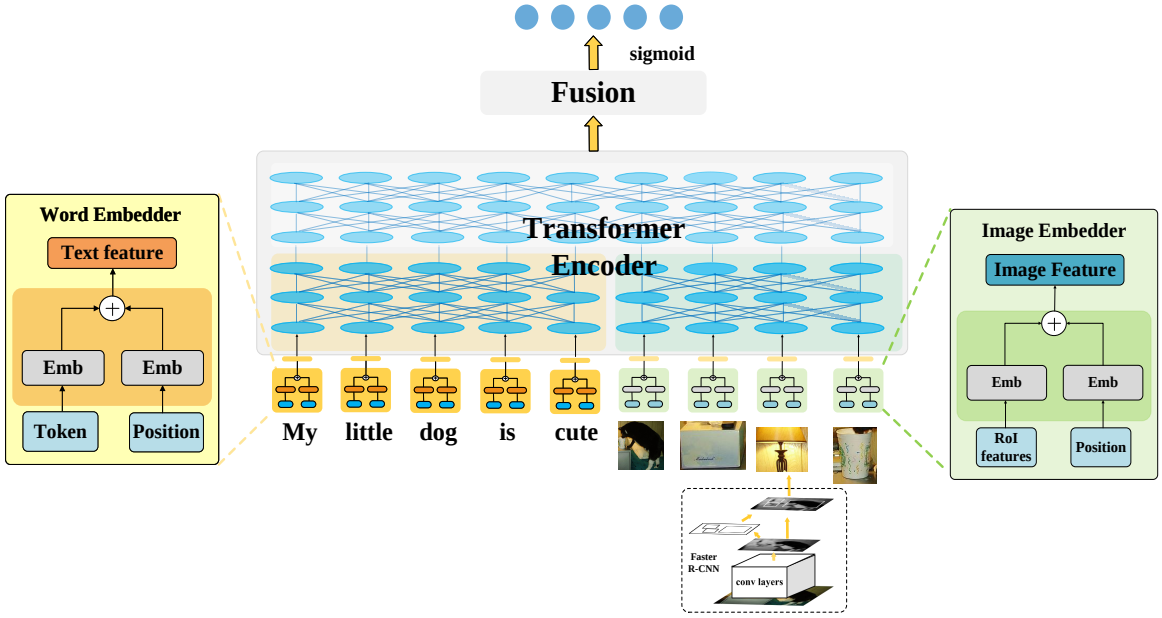

Figure 1: Overview of the proposed model specifically shows the general structure of Faster R-CNN and the details of the embedders.

modality. For the multi-modal task of text and image, the previous practice is to input the text and image into two different pre-training models for processing the text and image modalities respectively, and then concatenate the output features and predict the emotion. However, this method lacks the processing of the alignment relationship between modalities. The proposed model considers the above three problems in the multi-modality field. Inspired by LXMERT, we use it as the main framework of our model. We use Faster R-CNN (Ren et al., 2017) to extract image RoI features and their position. For texts, we use BERT to extract text embedding. Then our system uses LXMERT (Tan and Bansal, 2019) to deal with the multi-modal alignment of text and image. After when two modalities are processed by LXMERT, we use the learnable integration mechanism Bilinear-interaction layer to fuse these features.

The remainder of this paper is organized as follows. In section 2, we described LXMERT and our fusion method in detail. The experimental results are presented in section 3. Finally, a conclusion is drawn in section 4.

2 System Overview

Task A and Task B are very similar in model structure except for the output layer. Therefore, we introduce the model we proposed as a whole. This model can be divided into four parts. They are the embedding layer for image and text preprocessing, the encoder for multi-modal presentation and alignment, the feature fusion layer, and the final output layer. The proposed model is as shown in Figure 2.

2.1 Embedding

For images, LXMERT does not simply use a convolutional neural network to output feature map but uses (Anderson et al., 2018) to extract objects from images. The image processing of LXMERT is similar to text processing inspired by BERT. The specific idea is to use Faster R-CNN to select 36 RoI (region of interest) boxes with high confidence for each image and use these boxes as the features of the image. Similar to the text processing of BERT, the model also considers the position of each box and embeds the corresponding position. 36 objects are extracted by Faster R-CNN as $\{o_1, \ldots, o_{36}\}$. f_j is the 2048 dimension RoI features of o_j, and p_j is its position. As is shown in figure 2, the processing of these variables is as follows:

$$\begin{aligned} \hat{f}_j &= \text{LayerNorm}\ (W_{\text{F}} f_j + b_{\text{F}}) \\ \hat{p}_j &= \text{LayerNorm}\ (W_{\text{P}} p_j + b_{\text{P}}) \\ v_j &= \left(\hat{f}_j + \hat{p}_j\right)/2 \end{aligned} \tag{1}$$

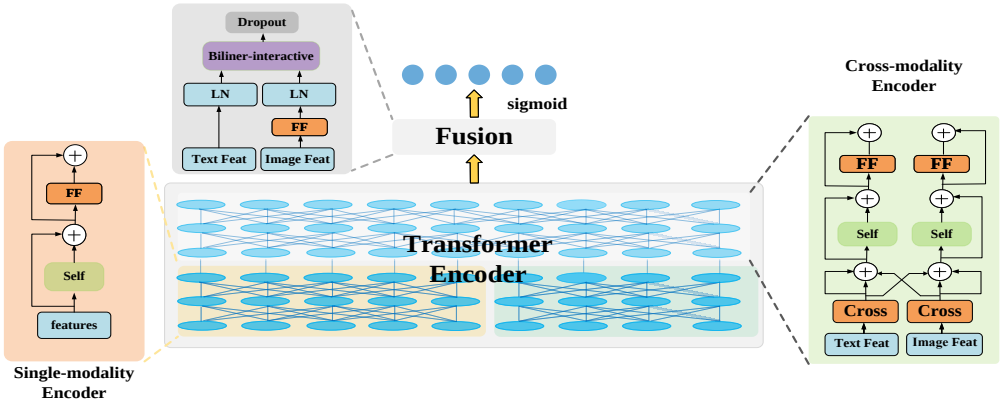

Figure 2: Overview of the proposed model, which specifically shows the details of the encoders and fusion layer. 'Self' and 'Cross' represent self-attention sub-layers and cross-attention sub-layers, respectively. 'FF' denotes a feed-forward sub-layer.

where W_F and W_P are the trainable weights of fully connected layer in matrix format. Moreover, b_F and b_P are the bias of the layer. $\hat{f}_j$ and $\hat{p}_j$ are the output of the layer-normalization.

For the text, sentences are converted into tokens whose length is equal to the length of scent according to the practice of WordPiece tokenizer (Wu et al., 2016). For instance, when the length of the sentence is n, the word tokens are $\{w_1, ..., w_n\}$. Then word w_i and its index i (the absolute position of w_i) are projected to vectors by embedding sublayers.The specific structure of embedder is shown in Figure 1. Then added to the index-aware word embedding:

$$\begin{aligned} \hat{w}_i &= \text{WordEmbed}\ (w_i) \\ \hat{u}_i &= \text{IdxEmbed}\ (i) \\ h_i &= \text{LayerNorm}\ (\hat{w}_i + \hat{u}_i) \end{aligned} \tag{2}$$

The specific structure of embedder is shown in Figure 1.

2.2 Attention layer

In this subsection, we will give a brief description of the attention mechanism. The principle of the attention mechanism is to give a request vector x and its context vector y_j, then, calculate the correlation between x and each y_j, and get a correlation score. The correlation score used in LXMERT is the dot product of vector x and vector y_j. After calculating the scores of all relevant context vectors y_j for x, LXMERT uses softmax to convert each score into a probability α_j to obtain the attention distribution.

$$\begin{aligned} a_j &= \text{score}\,(x, y_j) \\ \alpha_j &= \exp\,(a_j) \,/\, \sum_k \exp\,(a_k) \end{aligned} \tag{3}$$

$$\text{Att}_{\text{X}\to\text{Y}}\,(x, \{y_j\}) = \sum_j \alpha_j y_j \tag{4}$$

The output of the layer is the weighted sum of all probabilities with y_i.

The self-attention layer in LXMERT is implemented in a similar way to the attention layer, except that the query vector x in self-attention comes from the context-dependent vector y_i.

2.3 Encoder

The processing of image modality and text modality is shown in Figure 2. After embedding two modalities, LXMERT uses the two transformer single-modality encoders. One is a text encoder and another is an image encoder. Each layer in a single-modality encoder contains a self-attention ('Self') sub-layer and a feed-forward ('FF') sublayer, where the feed-forward sub-layer is further composed of two fully-connected sub-layers. We take N_L and N_R layers in the language encoder and the object-relationship encoder, respectively. We add a residual connection and layer normalization (annotated by the '+' sign in Figure 2) after each sub-layer as in Transformer (Vaswani et al., 2017). The features processed by a single-modality encoder will be first sent to another encoder called the cross-modality layer. Its main function is to align the features of the two modalities. The bi-directional cross-attention sublayer

Emotion category	Number of label '1'	Overall proportion
misogynous	5000	50.00%
shaming	1274	12.74%
stereotype	2810	28.10%
objectification	2202	22.02%
violence	953	9.53%

Table 1: Training dataset label analysis.

contains two unidirectional cross-attention sublayers, one from image to text and the other from text to the image. LXMERT stacks them N_x times, the input of k-th layer is the output of the previous$(k-1)$-th layer. Similarly, the query and context vectors are the outputs of the $(k-1)$-th layer. The method of processing the text features h_i^{k-1} and the image features v_j^{k-1} in unidirectional cross-attention sub-layers is as follows:

$$\begin{aligned} \hat{h}_i^k &= \text{Cross } Att_{\text{L}\to\text{R}}\left(h_i^{k-1}, \left\{v_1^{k-1}, \ldots, v_m^{k-1}\right\}\right) \\ \hat{v}_j^k &= \text{Cross } Att_{\text{R}\to\text{L}}\left(v_j^{k-1}, \left\{h_1^{k-1}, \ldots, h_n^{k-1}\right\}\right) \end{aligned} \tag{5}$$

where $\hat{h}_i^k$ and $\hat{v}_j^k$ are the output of the cross-attenton layer.

Then LXMERT further inputs the features processed by the cross-modality sublayer to the self-attention sublayer. This method aimed to further construct the internal connection of each modality after alignment. The specific treatment is:

$$\begin{aligned} \tilde{h}_i^k &= \text{Self } Att_{\text{L}\to\text{L}}\left(\hat{h}_i^k, \left\{\hat{h}_1^k, \ldots, \hat{h}_n^k\right\}\right) \\ \tilde{v}_j^k &= \text{Self } Att_{\text{R}\to\text{R}}\left(\hat{v}_j^k, \left\{\hat{v}_1^k, \ldots, \hat{v}_m^k\right\}\right) \end{aligned} \tag{6}$$

$\hat{h}_i^k$, $\hat{v}_j^k$ then processed by self-attention to $\tilde{h}_i^k$ and $\tilde{v}_j^k$, which will be further input to an 'FF' sublayer, connected through a residual, and input to the normalization to obtain the final output h_i^k, v_j^k. For each text in the data, the model will generate a Pooler output. We use the Pooler of each sentence as the output of the text modality.

2.4 Fusion

The method of this layer is inspired by Sina's paper FiBiNET by Huang et al. (2019). After LXMERT outputs two modality features, we need to further process its output. The dimension of image features is 36×768, while the dimension of text features is 768. To better integrate the two modalities, we flatten the image features and change its dimension to 768 through a feed-forward layer. Then, each modality will be normalized through layer normalization. Then, the features of each modality are sent to the Bilinear-interactive layer.

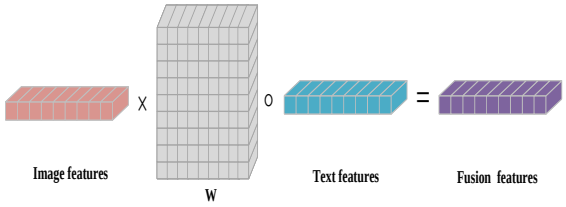

Figure 3: Bilinear-interactive layer.

The idea of the Bilinear-interactive layer is as shown in Figure 3. We establish a k-order square matrix W, which is trainable. To fuse the information of various modalities, 768-dimensional image features will first inner product with W. Then, for text features, we use Hadamard product to multiply the previous matrix. We finally use the dropout layer to improve the generalization ability of the model.

2.5 Output layer

- Sub-task A: this task is a binary classification task, so in the output layer, we use a shape of 768×1 full connection layer and use sigmoid as the activation function to process the results.

- Sub-task B: this task is a multi-label classification task. Therefore, in the output layer, we use a full connection layer whose shape is 768×5. Since each label classification is equivalent to binary classification, we use sigmoid as the activation function to process the results during output.

3 Experiments and Evaluation

3.1 Dataset

The task organizer provided 10000 pieces of data for training, including meme images with image serial numbers and text descriptions corresponding to the image. In the training dataset, there

are 10000 images and an excel table to record the text corresponding to the images and supervise the learning of the corresponding labels.

When analyzing the data, we found that different labels account for different proportions in the number of their respective classifications. For the misogynous tag, both 0 and 1 categories account for 50%, so the data sample tag is more balanced for a supervision task. However, for the other four labels such as sharing, the proportion of label 1 is only 12.74%. Among 10000 samples, the label of violence accounts for only 9.53%. Table 1 shows the proportion of each label in the training dataset. As shown in Table 1, we find that the proportion of labels of different categories is very different, and there is data imbalance. This will make the model have a strong learning effect on a large classification label and easy to classify. However, for the low proportion of classification tags, it is difficult to learn and classify.

Based on this, we use Focal loss by (He et al., 2016) as the loss function of our model.

3.2 Experimental configuration

Our model is based on TensorFlow platform version 2.5.0. The main model adopts LXMERT from the Hugging Face transformers toolkit. We first use*UNC-NLP/LXMERT-base-uncased* tokenizer-Fast to process our text to embeddings, and we also use *UNC-NLP/LXMERT-base-uncased* pretrained model as our base model LXMERT's pretrained model. The Adam optimizer (Kingma and Ba, 2015) was used to update all trainable parameters. The Hyper-parameters configuration used in the model is shown in Table 2: We use Faster R-CNN to extract features of images, which is based on the paper by (Anderson et al., 2018). In this task, we use an open-source docker image *airsplay / bottom-up attention* and use a Faster R-CNN pretraining model based on ResNet101 to extract 36 RoI feature boxes and their corresponding position.

3.3 Evaluation Metrics

Sub-task A Systems will be evaluated using macro-average F1-score. In particular, for each class label (i.e. misogynous and not misogynous) the corresponding F1-score will be computed, and the final score will be estimated as the arithmetic mean of the two F1-score. Sub-task B Systems will be evaluated using weighted-average F1-score. In particular, the F1-score will be computed for each label and then their average will be weighted by support, i.e, the number of true instances for each label.

Adam Optimizer config	**Value**
Learning rate	5e-5
epsilon	1e-8
Focal loss config	**Value**
alpha	0.25
gamma	3
batch size	16
epoch	20

Table 2: Hyper-parameters config.

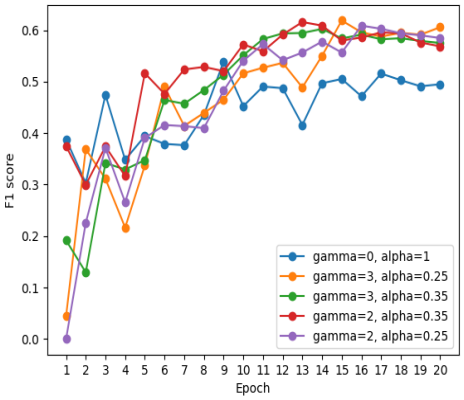

Figure 4: The ablation experiment of the Focal loss for different hyperparameter

$$\begin{aligned} \text{precision} &= \frac{TP}{TP+FP} \\ \text{recall} &= \frac{TP}{TP+FN} \end{aligned} \tag{7}$$

TP is the number of true positives classified by the model. FN is the number of false negatives classified by the model. FP is the number of false positives classified by the model.

$$\text{F}_1\text{-score} = \frac{2 \times \text{ precision } \times \text{ recall}}{\text{precision } + \text{ recall}} \tag{8}$$

F_1-score is the harmonic average of recall and precision.

3.4 Hyperparametric selection

In this section, we mainly introduce the hyperparametric selection of focal loss of the model. We adjust the two hyperparameters γ and α of Focal loss and train the model. In the training dataset, we randomly take 90 % data for the training model and the remaining 10 % as the test dataset to test

Model	Task A F1-score	Task B F1-score
Only image(Base line)	0.639	N/A
Only text(Base line)	0.640	N/A
Only ELECTRA	N/A	0.5454
Only ResNet	N/A	0.4581
ELECRTA and ResNet with concatenate	N/A	0.4816
ELECRTA and and ResNet with Fusion	N/A	0.5041
Image and Text(Base line)	0.650	0.621
LXMERT without Fusion	0.655	0.629
LXMERT with Fusion	**0.662**	**0.633**

Table 3: Performance comparison of different models. As shown in the table, our proposed model achieves the best results.

the performance of the model. The loss function focal loss is modified based on the standard cross-entropy loss.

This function can reduce the easy-to-classify samples so that the model can more focus on the samples that are difficult to classify in training. p_t in cross-entropy loss function reflects the recognition ability of the model to this sample (i.e. how well the knowledge is mastered). We define p_t is:

$$p_{\mathrm{t}} = \begin{cases} p & \text{if } y = 1 \\ 1 - p & \text{otherwise} \end{cases} \tag{9}$$

The smaller the p_t is, the more difficult it is to classify, so contribution should be improved to the loss function when calculating the loss. Therefore, the specific method of Focal loss is to multiply a weight with p_t before the entropy loss function. α is balancing factor,$\alpha \in [0, 1]$, γ is modulating factor, $\gamma \in [0, 5]$. The Focal loss is as:

$$Focal\_loss(p_t) = -\alpha(1 - p_t)^{\gamma} log(pt) \tag{10}$$

Thus, when $\alpha = 1$, $gamma = 0$, focal loss is similar to the cross-entropy loss function. By changing the values of γ and α, we found that when $\alpha = 0.25$ and $\gamma = 3$, for sub-task B, the weighted F_1 score of our model reached 0.662 and 0.633. See Figure 4.

3.5 Model comparison

We compare our model to a baseline and a model that combines two pre-trained models based on ELECTRA (Clark et al., 2020) and ResNet-101 (Ren et al., 2017) in this section. ELECTRA deals with text modality and ResNet is used to deal with image modality. The methods of feature fusion are compared with the Bilinear-interactive layer and concatenate layer using direct concatenate. The specific task is based on sub-task B. See Table 3 for details.

4 Conclusion

In this task, we design an image and text multi-modality model based on LXMERT for multi-modality representation and alignment, and modality fusion based on the Bilinear-interaction layer. Compared with the traditional method of stitching two pre-training models for each modality then concatenating two features to predict emotion, this model considers the representation, alignment, and fusion of multi-modality, and achieves better results than the baseline method.

At the same time, we found that after adding the Bilinear-interaction layer, the performance of the model is better than using only feature concatenate. See Table 3. Meanwhile, when analyzing the data, we found that the background of the meme graph and some characters in the graph were not used as the target input model by Faster R-CNN, which may affect the accuracy of the model. Meanwhile, the size of the meme image is too small to include multiple targets, and the target is relatively single, which may affect the performance of the model.

5 Acknowledgement

This work was supported by the National Natural Science Foundation of China (NSFC) under Grants No. 61966038. The authors would like to thank the anonymous reviewers for their constructive comments.

References

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 6077–6086. Computer Vision Foundation / IEEE Computer Society.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: pre-training text encoders as discriminators rather than generators. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Elisabetta Fersini, Francesca Gasparini, Giulia Rizzi, Aurora Saibene, Berta Chulvi, Paolo Rosso, Alyssa Lees, and Jeffrey Sorensen. 2022. SemEval-2022 Task 5: Multimedia automatic misogyny identification. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*. Association for Computational Linguistics.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society.

Tongwen Huang, Zhiqi Zhang, and Junlin Zhang. 2019. Fibinet: combining feature importance and bilinear feature interaction for click-through rate prediction. In *Proceedings of the 13th ACM Conference on Recommender Systems, RecSys 2019, Copenhagen, Denmark, September 16-20, 2019*, pages 169–177. ACM.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Gen Li, Nan Duan, Yuejian Fang, Ming Gong, and Daxin Jiang. 2020. Unicoder-vl: A universal encoder for vision and language by cross-modal pre-training. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 11336–11344. AAAI Press.

Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. 2019. Visualbert: A simple and performant baseline for vision and language. *CoRR*, abs/1908.03557.

Wei Li, Can Gao, Guocheng Niu, Xinyan Xiao, Hao Liu, Jiachen Liu, Hua Wu, and Haifeng Wang. 2021. UNIMO: towards unified-modal understanding and generation via cross-modal contrastive learning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 2592–2607. Association for Computational Linguistics.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 13–23.

Bo Peng, Jin Wang, and Xuejie Zhang. 2020. Adversarial learning of sentiment word representations for sentiment analysis. *Information Sciences*, 541:426–441.

Di Qi, Lin Su, Jia Song, Edward Cui, Taroon Bharti, and Arun Sacheti. 2020. Imagebert: Cross-modal pre-training with large-scale weak-supervised image-text data. *CoRR*, abs/2001.07966.

Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. 2017. Faster R-CNN: towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(6):1137–1149.

Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. 2020. VL-BERT: pre-training of generic visual-linguistic representations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Hao Tan and Mohit Bansal. 2019. LXMERT: learning cross-modality encoder representations from transformers. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 5099–5110. Association for Computational Linguistics.

Junfeng Tian, Min Gui, Chenliang Li, Ming Yan, and Wenming Xiao. 2021. Mind at semeval-2021 task 6: Propaganda detection using transfer learning and multimodal fusion. pages 1082–1087. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Jin Wang, Liang-Chih Yu, K. Robert Lai, and Xuejie Zhang. 2019. Investigating dynamic routing in tree-structured lstm for sentiment analysis. pages 3430–3435. Association for Computational Linguistics.

Jin Wang, Liang Chih Yu, K. Robert Lai, and Xuejie Zhang. 2020. Tree-structured regional cnn-lstm model for dimensional sentiment analysis. *IEEE/ACM Transactions on Audio Speech and Language Processing*, 28.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144.

Li Yuan, Jin Wang, and Xuejie Zhang. 2020. Ynu-hpcc at semeval-2020 task 8: Using a parallel-channel model for memotion analysis. pages 916–921. International Committee for Computational Linguistics.

Xingyu Zhu, Jin Wang, and Xuejie Zhang. 2021. YNU-HPCC at semeval-2021 task 6: Combining ALBERT and text-cnn for persuasion detection in texts and images. In *Proceedings of the 15th International Workshop on Semantic Evaluation, SemEval@ACL/IJCNLP 2021, Virtual Event / Bangkok, Thailand, August 5-6, 2021*, pages 1045–1050. Association for Computational Linguistics.