Reasoning about Intermodal Correspondences

Mariia Podguzova*
st165056@stud.uni-stuttgart.de
University of Stuttgart
Germany

Son Tung Nguyen*
st165140@stud.uni-stuttgart.de
University of Stuttgart
Germany

ABSTRACT

Cross-modal tasks imply retrieval of modality-invariant feature representations, which are aligned and able to substitute or complement each other. This work is focused on finding of intermodal correspondences between textual and visual modalities with incorporation gaze information as a regularization technique. We use a student-teacher-model-like network, which is modality specific on the student level and modality invariant on the teacher level. Resulting aligned representations can be valuable for computer vision applications such as visual question answering, image captioning, person search. We also compare different sampling algorithms to explore their influence on the estimated results.

KEYWORDS

Computer vision, intermodal correspondences; gaze regularization; natural language processing; neural networks.

ACM Reference Format:
Mariia Podguzova and Son Tung Nguyen. 2020. Reasoning about Intermodal Correspondences. In *Proceedings Fachpraktikum Interaktive Systeme (FIS'20).* ACM, New York, NY, USA, 7 pages.

1 INTRODUCTION

Nowadays, media data represents a valuable source of information not only for human beings but also for technologies, especially for artificial intelligence. These technologies requires effective methods for processing and extracting the most significant information from multi-modal data to perform necessary tasks. However, usually, technologies are focusing on a particular problem and data type.

Since different modalities have different representations of features, there is a modality gap that needs to be closed in order to enable estimation of representation similarity. Therefore, the goal of this work is an implementation of a modal that forms representations aligned across modalities and reasoning about correspondences between textual, visual and gaze information.

Contribution. Our contribution is obtaining of a modality-invariant representation for textual and vision modalities and studying how gaze information can be beneficial for the cross-modal task. We also compare various sampling algorithms and explore their influence on the performance.

Organization. The rest of this paper is organized as follows. In Section 2, we describe datasets and modalities used in the work. In Section 3, we present a cross-modal deep network to obtain aligned representations. In Section 4, we explore training algorithm, possibilities for its modifications in sense of sampling algorithms. We discuss as well the way how we incorporate gaze information in our model. In Section 5, we describe conducted experiments and present some examples for result examination.

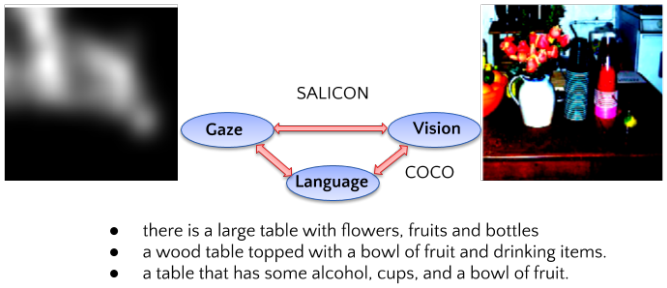

Figure 1: Correspondences between visual, textual and gaze information.

1.1 Related Works

Language and Vision. The earliest works related to image-text correspondences explore image-captioning task [9, 17]. These models based on combination of deep convolution and recurrent neural networks build natural language descriptions of the image content. In this work rather than generate sentences we aim to learn representations that are aligned across modalities. Y. Aytar et.al. [1] considered a similar task to find correspondences between text, images and audio. N. Sarafianos et.al. [14] examined the problem of text-to-image matching via adversarial representation approach based on BERT [2] and ResNet [6] networks.

Vision and Attention. Attention is the state-of-the-art approach in deep learning, which is inspired by human attention process and aims to improve performance of models. Attention mechanism is widely used for computer vision tasks. Attention maps or class activation mapping (CAM) [23] could be obtained in each category using the response on convolution layer. H. Fukui et.al. [3] proposed Attention Branch Network (ABN), which forms a branch structure with an attention mechanism. Attention can be beneficial in image-captioning and visual question answering tasks [21, 22].

Language and Attention. Although alignment between text and attention is not trivial task, multiple works consider this correspondences for high-level problems. F. Wang et.al. [19] proposed deep Decoupled Attention Network is applied for text recognition task. It uses jointly the feature maps and attention maps to solve alignment problem. D. Tuckey et.al. [15] presented the approach of generation salience maps in sequence-to-sequence model with bidirectional LSTM for text summarizing. Attention is also can be considered for explanation of a reading process [4]. In our work we do not use text-gaze pairs for training. Instead, we apply available attention information as regularization.

*Both authors contributed equally to this research.

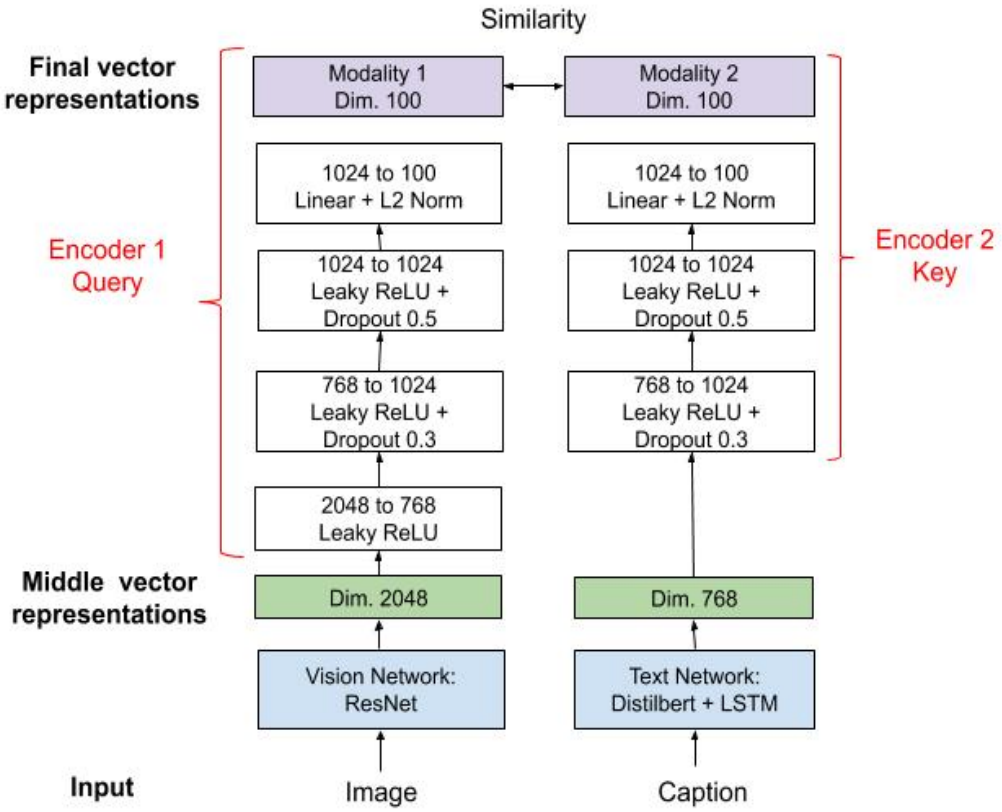

Figure 2: Architecture of the implemented network.

2 CROSS-MODAL NETWORK

In this section we describe datasets and modalities as well as architecture of our model.

We introduce the model that accepts as an input either an image or a sentence and produces feature representations that are aligned. One can formulate this task as follows. Given x_i and y_i samples from modalities x and y, respectively. In our case, x_i represents an image and y_i represents a sentence. Our goal is learning of representations that are aligned across modalities. Let $f_x(x_i)$ be the representation in modality x, and $f_y(y_i)$ be the representation in modality y. These two representations are aligned if they are close to each other in sense of some distance metric.

2.1 Datasets and Modalities

In this work two datasets are used: Microsoft Common Objects in Context (COCO) [10] to learn aligned representations for vision and language modalities and Saliency in Context(SALICON) [8] to incorporate gaze information. The examples of used data are shown in the Figure 1.

Images. COCO is a large-scale object detection, segmentation, and captioning dataset. It includes 2.5 million labeled instances in 328 thousand images of different sizes.

Our training dataset contains 10000 images while validation dataset includes 5000 images, which are transformed. In order to unify the images we resize them to the format 224x224, horizontally flip image tensors with the probability 0.5. We also normalize them with mean and standard deviation to get the data within a range and ensure faster convergence.

Text. COCO dataset also includes a description of each image, which is composed of five captions per image. We use these captions to collect the dataset for the textual modality. The numbers of sentences in training and validation datasets are the same as for images.

Each input token is transformed to the vector made up of 768 float numbers that are accepted by 1-layer LSTM network. The dimension of the output layer is also equal to 768. As data processing step we perform padding shorter sentences with the token id 0 to the length of a longest sentence in the dataset.

Gaze. SALICON dataset offers a large set of saliency annotations on the COCO dataset. Although it allows only using a general-purpose mouse instead of eye tracker to record viewing behaviors, we have opportunities to find correspondences between images and attention maps, which were transformed in the same way as images.

2.2 Architecture

The network has a teacher-student model-like architecture with student networks included vision and text networks as a bottom

layer and teacher networks included two encoders as a top layer, see Figure 2. Student networks accept as inputs either an image or text and produces model specific representations that are used by teacher networks to output representations that are aligned across these two modalities.

Vision Network. As the vision network we use a deep residual network, which has shown excellent results in COCO detection and COCO segmentation tasks [6]. The basic idea of this type network is using a residual mapping $F(x) := H(x) - x$, which is easier to optimize, where $H(x)$ is an underlying mapping.

In our case the variation with 50 layers is applied. It is constructed from 3-layer bottleneck blocks on each layer. Outputs of this modality are the intermediate vector representations with a dimension 2048, see Table 1.

Text Network. For implementation of the textual model we combine pre-trained DistilBERT embeddings [13] and recurrent LSTM network.

DistilBERT is a distilled version of BERT [2] network based on a compression technique in which a compact student model is trained to reproduce the behaviour of a larger teacher model or an ensemble of models. K. He et.al. demonstrate the advantages of this pre-trained version of BERT for many NLP tasks.

Teacher Networks. The teacher level is presented by two encoders. The architectures are constructed by fully-connected layers with different number of parameters, leaky ReLU as an activation function and dropout for regularization. Encoder 1 accepts outputs from the vision network and produces vector representations of a dimension 100 invariant for two modalities. Analogically for encoder 2, which accepts outputs from the text network.

Layer name	50-Layer	Number of blocks
conv1	7x7,64,stride 2	
conv2	3x3, max pool, stride2	
	1x1,64 3x3,64 1x1,256	3
conv3	1x1,128 3x3, 128 1x1,512	4
conv4	1x1,256 3x3,256 1x1,1024	6
conv5	1x1,512 3x3,512 1x1,2048	3
average pool, 2048-d		

Table 1: Architecture of the vision network.

3 TRAINING ALGORITHM

We train our model with the contrastive unsupervised learning algorithm [7, 16].

Methods based on contrastive loss estimation build dynamic dictionaries. The keys of the dictionary are sampled from data, in our case images. Encoders are trained to perform dictionary look-up: an encoded query should be similar to its corresponding key and dissimilar to other. The goal of the learning process is minimizing a contrastive loss.

Thereby, contrastive losses do not match an input to a known target as for supervised learning tasks. Instead, they measure similarities of sample pairs in a representation space. In that case the target is varying and it depends on the computed data representations.

3.1 Noise Contrastive Estimation

The visual encoder (Teacher1) produces encoded image queries and the language encoder (Teacher2) forms sets of keys of the dictionary, see Figure 3. A set of negative keys $\{k_0, k_1, k_2, ...\}$ is denoted as *k_neg*. Assume there is a single positive key k_+ that has a correspondence q. A contrastive loss is a function, which has the lowest value when q is the most similar to the positive key k_+ and dissimilar to all other keys from the set of negative keys.

We define similarity measure as a dot product of two samples. Therefore, a contrastive loss is presented in the form of InfoNCE:

$$L_q = -log\frac{q \cdot k_+/\tau}{\sum_{i=0}^{K} exp(q \cdot k_i/\tau)}, \quad (1)$$

where τ is a hyper-parameter [16]. The sum is over one positive and K negative samples. This loss could be explained as the logistic loss of a K + 1-way softmax-based binary classifier that tries to classify a query q as a positive key k_+.

3.2 Sampling algorithms

The performance of the model varies depending on on how negative keys are sampled. We consider three algorithms: batch, queue and rerank sampling.

Batch sampling. Although InfoNCE is well-suited to train large dictionaries, it is not effective for batch mode training on GPUs. The main problem is using of different sets of negative samples. In that case, it is difficult to describe the learning process with dense matrix operations. As a consequences, the training time considerably increases. This problem can be solved by applying batches to share noise samples (Batch-NCE) [12]. It implies the following ideas:

- each vector in a batch can be used as a positive sample k_+;
- rest n-1 vectors can be considered as negative samples in the set *k_neg* for a batch size of n.

Queue sampling. To follow the idea of queue sampling we use Momentum Contrast algorithm [5]. Important features can be learned by a large dictionary with significant amount of negative samples, while the encoder for keys remains consistent. The dictionary is built as a queue, with the current mini-batch enqueued and the oldest mini-batch dequeued, decoupling it from the mini-batch size. The dictionary size can be larger than a mini-batch size and can be adjusted as a hyper-parameter. We define the following properties of this algorithm:

- maintains of a queue of negative samples;

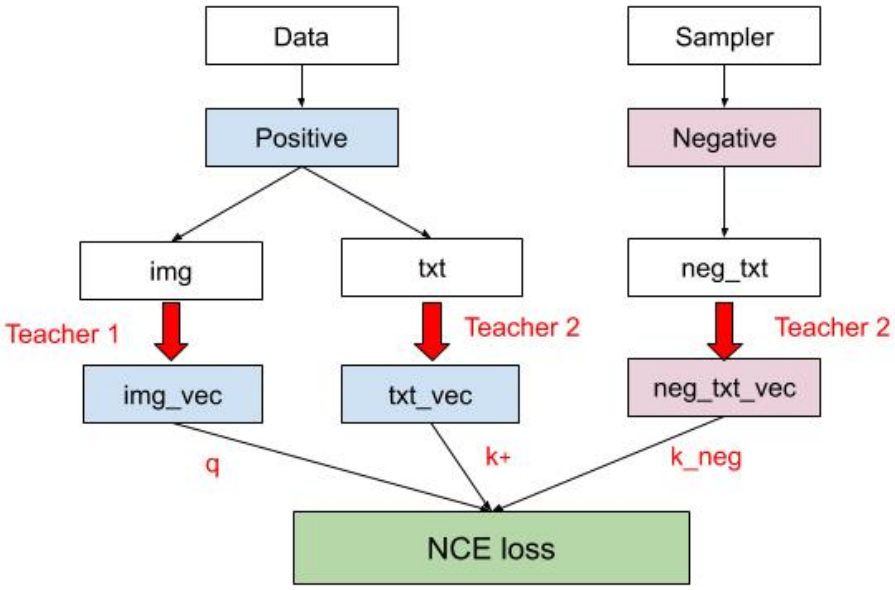

Figure 3: Training algorithm.

- on each iteration allows to push a batch into the queue or pop one batch from the queue;
- allows to control the amount of negative samples.

Rerank sampling. The main difference of rerank sampling from previous algorithms is scoring of all vectors in the space [11]. A sample obtains higher score, than it is more similar to the comparable one. For negative space only n vectors with highest score are sampled, n is a hyperparameter that should be tuned. We highlight the following characteristics of this sampling approach:

- scores all the vectors in the space;
- samples only n vectors with highest score;
- requires n large enough to be effective;
- computationally expensive when n is large.

3.3 Regularization with gaze information

To incorporate gaze information in our model we apply attention mechanism that is widely used in computer vision and natural language processing [18, 20]. This way, we enforce the network to focus on the most informative areas of the images.

We use element-wise multiplication of images and corresponding attention maps as an attention mechanism.

$$g'_c(x_i) = M(x_i) \cdot g_c(x_i), \tag{2}$$

where $g_c(x_i)$ is an input image, $g'_c(x_i)$ is the result of attention mechanism, $M(x_i)$is an attention map. Note that $\{c|1, .., C\}$ denotes the index of the channel. The examples of resulting inputs are shown in the Figure 4.

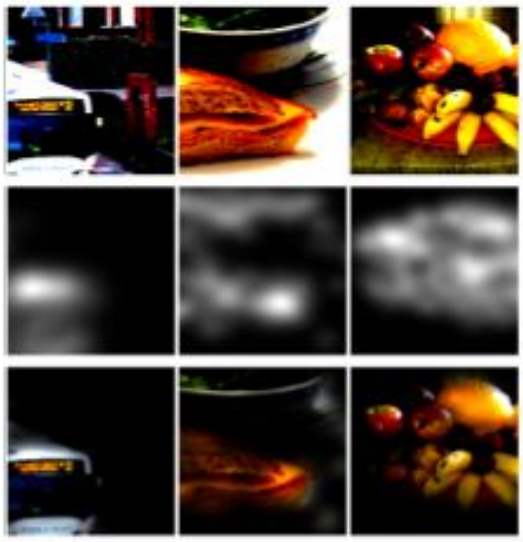

Figure 4: Attention mechanism: element-wise multiplication of images and corresponding attention maps. **First row:** images from COCO dataset. **Second row:** attention maps from SALICON dataset. **Third row:** the results of element-wise multiplication.

4 EXPERIMENTS

We train our model within 50 epochs using mini-batch stochastic gradient descent with the following parameters: learning rate 0.0002, weight decay 0.0001 and momentum 0.9. The training is

Batch size	Accuracy	Loss	Avg Similarity
Validation set			
16	0.595	1.213	0.028
32	0.602	1.180	0.031
48	0.610	1.140	0.038
64	0.617	1.146	0.034
Training set			
16	0.858	0.448	0.228
32	0.757	0.788	0.117
48	0.685	1.042	0.073
64	0.612	1.312	0.035

Table 2: Results for various batch sizes. Overfitting for batch sizes 16,32 and 48.

Queue size	Accuracy	Loss	Avg Similarity
Validation set			
0.3	0.561	1.276	0.020
0.5	0.574	1.232	0.023
0.7	0.593	1.208	0.026
1.0	0.610	1.141	0.035
Training set			
0.3	0.553	1.513	0.016
0.5	0.560	1.481	0.018
0.7	0.585	1.424	0.023
1.0	0.622	1.297	0.037

Table 3: Results for various queue sizes.

memory consuming and requires more than 10GB available memory space, therefore, two GPU GeForce GTX 1080 were used. It takes 6-8 hours depending on the type of experiment.

4.1 Results

First experiment: various batch sizes. In the first experiment we compare the results with various batch sizes, see Figure 5 and Table 2. The highest accuracy have been achieved with the batch size 64, while the smallest loss value does not correspond the best value of accuracy. These findings lead to assumption that other metrics could better evaluate the result.

Moreover, the comparison of the results for training and validation datasets indicates overfitting for the batch sizes 16,32 and 48. The trainning losses are significantly lower than the loss values for the validation set.

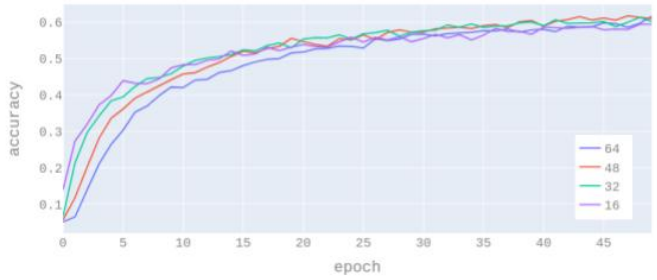

Figure 5: Validation accuracy with various batch sizes. **Purple:** 16. **Green:** 32. **Red:** 48. **Blue:** 64.

Second experiment: various queue sizes. In the second experiment we study the influence of various queue sizes. We consider 64 samples per batch but different proportions of the queue size, see Figure 7. The largest queue size corresponds the highest accuracy and the smallest loss value.

In comparison to batch sampling for the batch size 64 we obtain lower loss values, however, the accuracy in the previous case is slightly higher. This approach prevents the model from overfitting and improves its generalization. However, this approach does not ensure fast convergence.

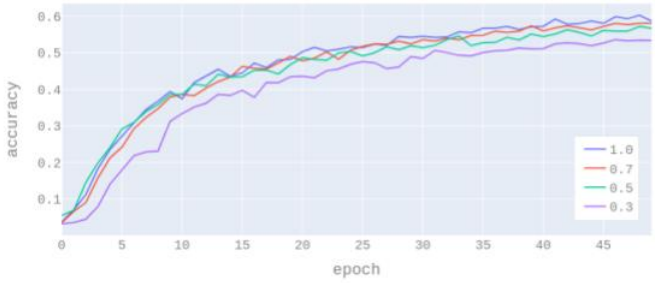

Figure 6: Validation accuracy with various queue sizes and batch size=64. **Purple:** 0.3. **Green:** 0.5. **Red:** 0.7. **Blue:** 1.0.

Third experiment: regularization with gaze. In the third experiment we incorporate gaze information using the attention mechanism described in Section 6. We train the model with batch sampling and apply different proportions of the gaze data. The results are shown in Table 4, the first raw with attention probability 0.0 corresponds to the training without sampling of attention maps.

Although, the pure batch sampling algorithm with batch 64 outperforms the algorithm with attention maps, batch sampling with smaller batch sizes does not allow us to obtain comparable results. Meanwhile,incorporating gaze information prevents overfitting for smaller batch sizes, see the results for batch size 48 in Table 5. We do not observe large differences between validation and training losses. Another advantages of this approach is that it provides faster convergence in comparison to others.

4.2 Examples

For result examination we present some examples how our model works. We give the network an image and obtain the ranking list of the most suitable captions from it.

Figure 8 presents the example when the network matches 3 captions to the image. Moreover, the network correctly determines the caption with rank 1 that is the ground truth.

Figure 9 shows more complicated case when the network matches 14 captions to the image. Ground truth has rank 14 but the captions with highest ranking is also can be interpreted as appropriate descriptions for given image.

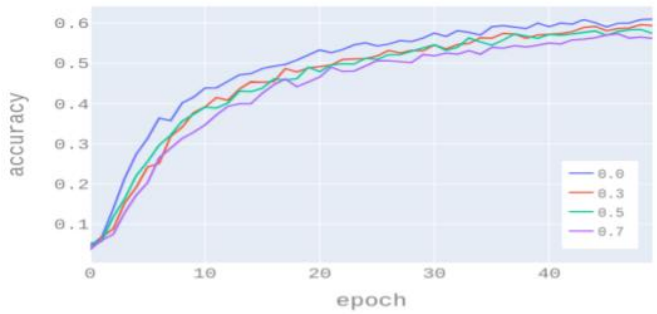

Figure 7: Validation accuracy with gaze regularization and batch size=64. **Purple:** 0.7. **Green:** 0.5. **Red:** 0.3. **Blue:** 0.0.

Attention	Accuracy	Loss	Avg Similarity
Validation set			
0.0	0.617	1.146	0.034
0.3	0.596	1.190	0.028
0.5	0.589	1.217	0.025
0.7	0.575	1.265	0.021
Training set			
0.0	0.612	1.312	0.035
0.3	0.585	1.428	0.024
0.5	0.574	1.418	0.023
0.7	0.554	1.500	0.015

Table 4: Results for various attention probabilities. The probability 0.0 corresponds to the batch sampling with batch size 64.

Attention	Accuracy	Loss	Avg Similarity
Validation set			
0.0	0.610	1.140	0.038
0.3	0.597	1.177	0.031
0.5	0.591	1.207	0.029
0.7	0.581	1.232	0.025
Training set			
0.0	0.685	1.042	0.073
0.3	0.648	1.161	0.056
0.5	0.653	1.162	0.055
0.7	0.632	1.206	0.051

Table 5: Results for various attention probabilities. The probability 0.0 corresponds to the batch sampling with batch size 48.

The inverse matching from text to the rank list of predicted images is also possible, see Figure 10.

5 CONCLUSION

In this work we introduce a deep network for learning modality-invariant representations for textual and visual inputs. We compare various sampling algorithms, which have influence on the performance of NCE, used as the loss function. The third modality i.e., gaze information, incorporated as regularization factor, prevents overfitting for batch sampling and provides faster convergence. In the future work we aim to find the most appropriate evaluation metrics and approximate rerank sampling that would require less computational cost. An evaluation of the similarity between representations for the same modality could be also beneficial.

1. a table topped with four plates filled with food
2. a table with several dishes of asian food
3. a white table with various plates and dishes of food.
4. this table is filled with a variety of different dishes.

Figure 8: Simple case: the network predicts the correct answer.

1. a pile of oranges in crates topped with yellow bananas.
2. a grocery store filled with lots of fresh produce.
3. a farmers market fulled of fresh fruits and vegetables.
4. a market area with various crates of vegetables.
5. a display at grocery store filled with fruits and vegetables next to jars.
6. a produced shelf in a store filled with fruits and veggies.
7. a red stand with various baskets of fruits next to bar.
8. a display at a grocery store filled with fruits and vegetables.
9. a market area with boxes of fruit that includes bananas and pineapples.
10. benches of fruit on display at a market
11. fresh fruits and vegetables sit for sale at a store.
12. a bunch of bananas are sitting on table at the market.
13. a store with lots of unripe bananas and other products.
14. there are many bananas on a counter at a market

Figure 9: More complicated case: the correct caption with rank 14.

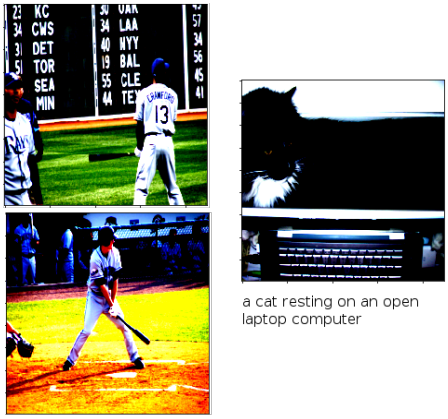

Figure 10: Left column: given caption corresponds to the image with the rank number 2. **Right column:** given caption corresponds to the image with the rank number 1.

We believe cross-modal representations provide valuable information for various high-level tasks and have a great potential to enable artificial intelligence perceive multi-modal data.

REFERENCES

[1] Yusuf Aytar, Carl Vondrick, and Antonio Torralba. 2017. See, Hear, and Read: Deep Aligned Representations. *Computing Research Repository* (2017).

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR* abs/1810.04805 (2018). arXiv:1810.04805 http://arxiv.org/abs/1810.04805

[3] Hiroshi Fukui, Tsubasa Hirakawa, Takayoshi Yamashita, and Hironobu Fujiyoshi. 2019. Attention Branch Network: Learning of Attention Mechanism for Visual Explanation. *Conference on Computer Vision and Pattern Recognition (CVPR)* (2019).

[4] Michael Hahn and Frank Keller. [n. d.]. Modeling Human Reading with Neural Attention. *Conference on Empirical Methods in Natural Language Processing (EMNLP), year = 2016* ([n. d.]).

[5] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2019. Momentum Contrast for Unsupervised Visual Representation Learning.

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. [n. d.]. Deep Residual Learning for Image Recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* ([n. d.]).

[7] Olivier J. Hénaff, Aravind Srinivas, Jeffrey De Fauw, Ali Razavi, Carl Doersch, S. M. Ali Eslami, and Aaron van den Oord. 2019. Data-Efficient Image Recognition with Contrastive Predictive Coding.

[8] Ming Jiang, Shengsheng Huang, Juanyong Duan, and Qi Zhao. 2015. SALICON: Saliency in Context. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2015).

[9] Andrej Karpathy and Fei-Fei Li. 2015. Deep visual-semantic alignments for generating image descriptions. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

[10] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft COCO: Common Objects in Context. (2014).

[11] Zhuang Ma and Michael Collins. 2018. Noise Contrastive Estimation and Negative Sampling for Conditional Models: Consistency and Statistical Efficiency. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

[12] Youssef Oualil and Dietrich Klakow. 2017. A Batch Noise Contrastive Estimation Approach for Training Large Vocabulary Language Models. *Computing Research Repository (CoRR)* (2017).

[13] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter.

[14] Nikolaos Sarafianos, Xiang Xu, and Ioannis A. Kakadiaris. 2019. Adversarial Representation Learning for Text-to-Image Matching.

[15] David Tuckey, Krysia Broda, and Alessandra Russo. 2019. Saliency Maps Generation for Automatic Text Summarization. *Computing Research Repository (CoRR)* (2019).

[16] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation Learning with Contrastive Predictive Coding. *Computing Research Repository (CoRR)* (2018). http://arxiv.org/abs/1807.03748

[17] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2014. Show and tell: A neural image caption generator. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2014).

[18] Fei Wang, Mengqing Jiang, Chen Qian, Shuo Yang, Cheng Li, Honggang Zhang, Xiaogang Wang, and Xiaoou Tang. 2017. Residual Attention Network for Image Classification. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2017).

[19] Tianwei Wang, Yuanzhi Zhu, Lianwen Jin, Canjie Luo, Xiaoxue Chen, Yaqiang Wu, Qianying Wang, and Mingxiang Cai. 2019. Decoupled Attention Network for Text Recognition.

[20] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. In *Proceedings of the 32nd International Conference on Machine Learning (Proceedings of Machine Learning Research)*.

[21] Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alexander J. Smola. 2015. Stacked Attention Networks for Image Question Answering. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2015).

[22] Quanzeng You, Hailin Jin, Zhaowen Wang, Chen Fang, and Jiebo Luo. 2016. Image Captioning with Semantic Attention. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016).

[23] Bolei Zhou, Aditya Khosla, Àgata Lapedriza, Aude Oliva, and Antonio Torralba. 2016. Learning Deep Features for Discriminative Localization. (2016).