Multi-modality Self-supervised learning

Paper Review #2

Multi-modality Self-sueprvision

Jong Hak Moon & Hyungyung Lee

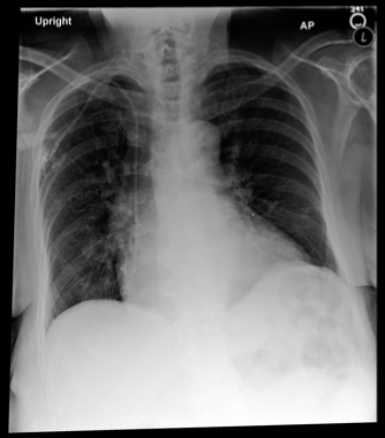
Aug. 6th. 2020

Recap

Free-text (DICOM Ver.)

Label (JPG Ver.)

Frontal



Lateral

- ✓ Data : Image (DICOM,JPG), Free-text reports(DICOM ver.), Labels (JPG ver. Chexpert & Negbio) [1:positively mentioned, 0: negatively mentioned, -1:ambiguous]
- ✓ DICOM Image (Pixel value over 255), JPG Image (Pixel value normalized 0-255)
- ✓ Dataset consists of 377,110 images corresponding to 227,827 radiographic studies, and electronic health record (EHR), In 47% (107,186 images) of all data, a single class is shown.
- ✓ Multimodal (using free-text & Image) deep learning via self-supervised approach

Recap

Key words

✓ 1. Multi-modal deep learning approach

✓ 2. Self-supervised approaches in vision

✓ 3. Self-supervised approaches in natural language

Related work survey <BERT based Self-supervision>

Clinical NLP

NAACL 19> Publicly Available Clinical BERT Embeddings

MLHC 20> CheXpert++: Approximating the CheXpert labeler for Speed, Differentiability, and Probabilistic Output

Arxiv 20> CheXbert: Combining Automatic Labelers and Expert Annotations for Accurate Radiology Report Labeling Using BERT

Multi-modality

NIPS 19> ViLBERT- Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks

ICLR 20> VL-BERT- PRE-TRAINING OF GENERIC VISUALLINGUISTIC REPRESENTATION

AAAI 20> Unicoder-VL: A Universal Encoder for Vision and Language by Cross-modal Pre-training

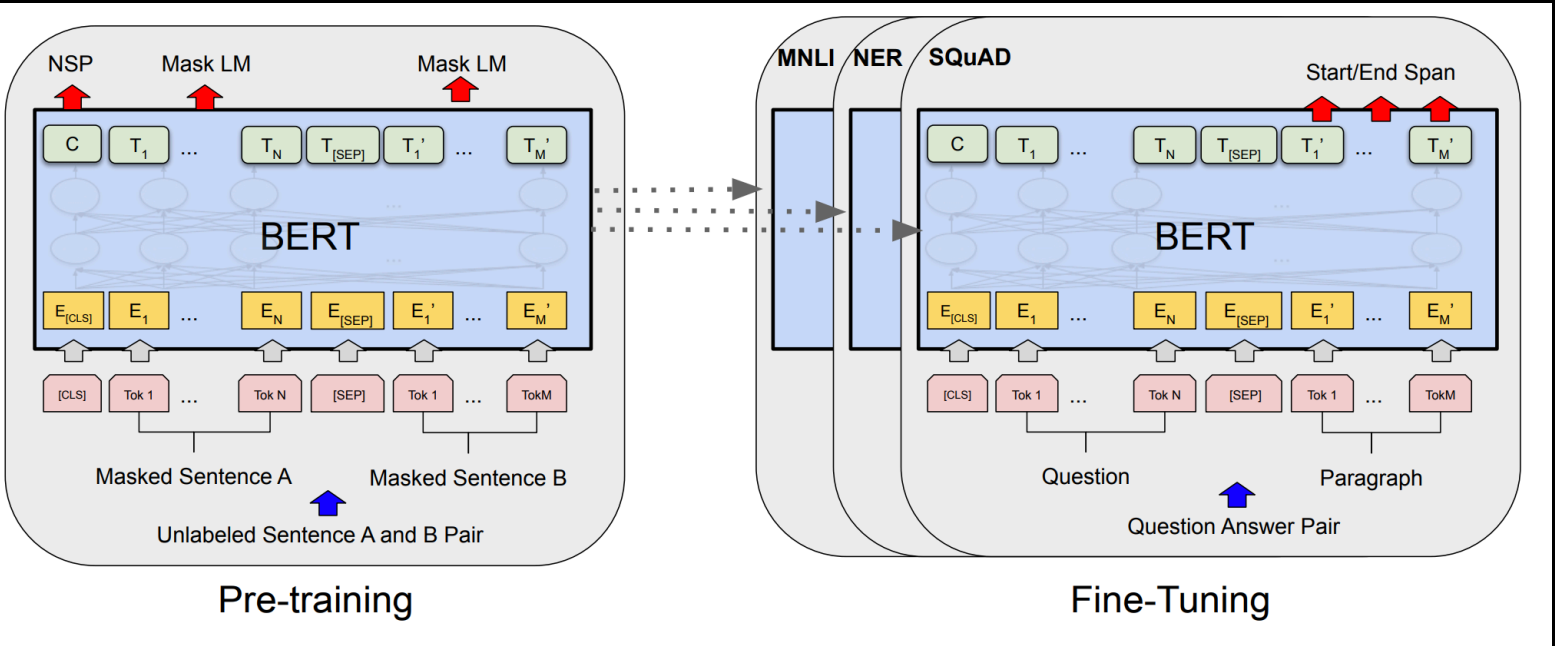
Methods for Self-supervised Joint representation learning

Related work survey <Multi-modal Self-supervision>

Method	Architecture	Visual token
VideoBERT <ICCV 19>	One single modal BERT + One single modal video token + One cross-modal transformer	Vector quantization (Video)
Contrative Bidirectional Transformer <Under Review>	One single modal BERT + One single modal CBT(Video) + One cross-modal CBT (1 or 0)	Output of a 3D CNN
ViLBERT <NeurIPS 19>	Two stream	Image RoI
InterBERT <Under Review>	Single stream & Two stream	Image RoI
Pixel-BERT <Under Review>	Single stream	Full Image

Recap

- BERT


NSP
Mask LM
Mask LM
C
T1
TN
T[SEP]
T1'
TM'
BERT
E[CLS]
E1
EN
E[SEP]
E1'
EM'
[CLS]
Tok 1
Tok N
[SEP]
Tok 1
TokM
Masked Sentence A
Masked Sentence B
Unlabeled Sentence A and B Pair
Pre-training
MNLI
NER
SQuAD
Start/End Span
Question
Paragraph
Question Answer Pair
Fine-Tuning

Recap

- BERT

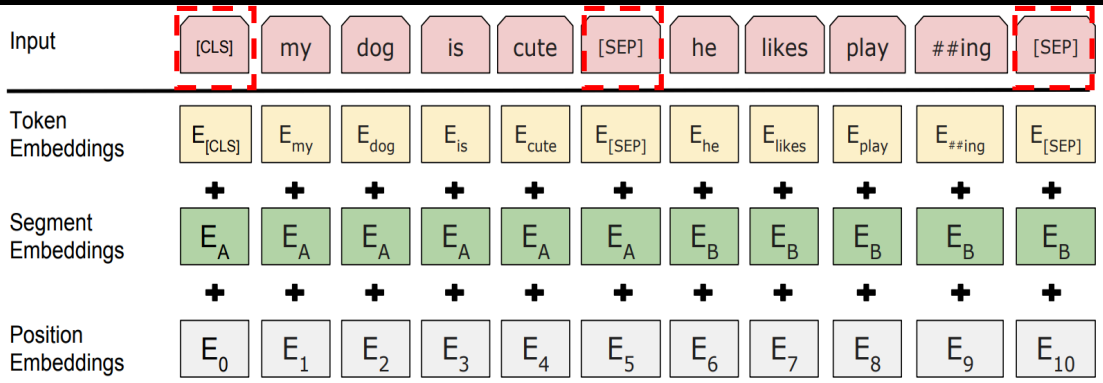


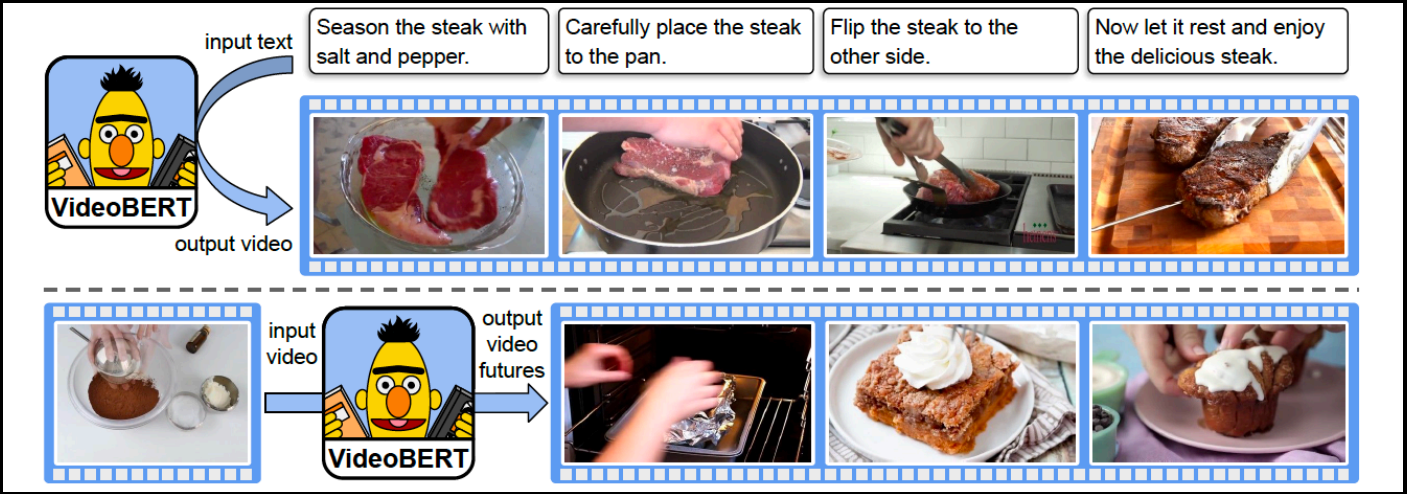
Figure 2: BERT input representation. The input embeddings are the sum of the token embeddings, the segmentation embeddings and the position embeddings.

Pre train task : #1 > Masked LM, #2 > Next Sentence Prediction (NSP)

$BERT_{BASE}$ (L=12, H=768, A=12, Total Parameters=110M)

$BERT_{LARGE}$ (L=24, H=1024, A=16, Total Parameters=340M).

1> VideoBERT: A Joint Model for Video and Language Representation Learning



- Objective : Apply BERT to learn a model of the form p(x|y), where x is a sequence of “visual words”, and y is a sequence of spoken words
- Pre-training : Mask Completion(forecasting), Linguistic-visual alignment classification(text 2 video)
- Downstream task : Zero-shot action classification, Video Captioning
- Main Contribution : Simple way to learn high level video representations that capture semantically meaningful and temporally long-range structure.

Method - Video BERT

1. Automatic speech recognition (ASR) system to convert sepeech into text from instructional videos (cooking videos).

2. Vector quantization (VQ) applied to low-level spatio-temporal visual features derived from pre-trained video classification models.

3. BERT model for learning joint distributions over sequences of discrete tokens. Model *P(x,y)*, where *x* is a sequence of visual words, and *y* is a sequence of spoken words.

Method – Video and Language Preprocessing

Video tokenize "Visaul words"

- Create clips from 30-frame (20fps, 1.5 sec)
- ConvNet (S3D) to extract features then 3D AvgPool (1024-D feature vector)
- Vector quantization : Tokenize the visual features using hierarchical k-means ($12^4 = 20736$ clusters)

We tokenize the visual features using hierarchical k-means. We adjust the number of hierarchy levels d and the number of clusters per level k by visually inspecting the coherence and representativeness of the clusters. We set $d = 4$ and $k = 12$, which yields $12^4 = 20736$ clusters in total. Figure 4 illustrates the result of this "vector quantization" process.

Language tokenize

- Automatic speech recognition (ASR) : To obtain text from the videos. Standard text preprocessing steps from BERT.
- Tokenize the text into WordPieces. (30,000 tokens)

Method – Video and Language Preprocessing

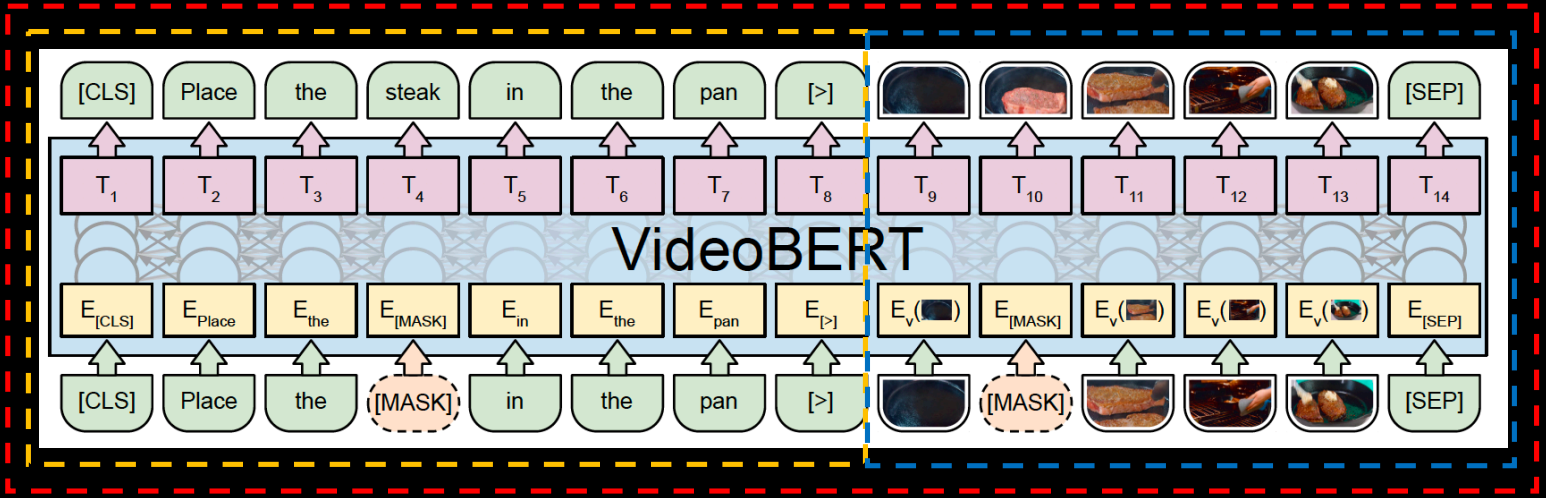
Result of Video tokenize “Visaul words”

“but in the meantime, you're just kind of moving around your cake board and you can keep reusing make sure you're working on a clean service so you can just get these all out of your way but it's just a really fun thing to do especially for a birthday party.”

“apply a little bit of butter on one side and place a portion of the stuffing and spread evenly cover with another slice of the bread and apply some more butter on top since we're gonna grill the sandwiches.”

Figure 4: Examples of video sentence pairs from the pretraining videos. We quantize each video segment into a token, and then represent it by the corresponding visual centroid. For each row, we show the original frames (left) and visual centroids (right). We can see that the tokenization process preserves semantic information rather than low-level visual appearance.

Method – VideoBERT model



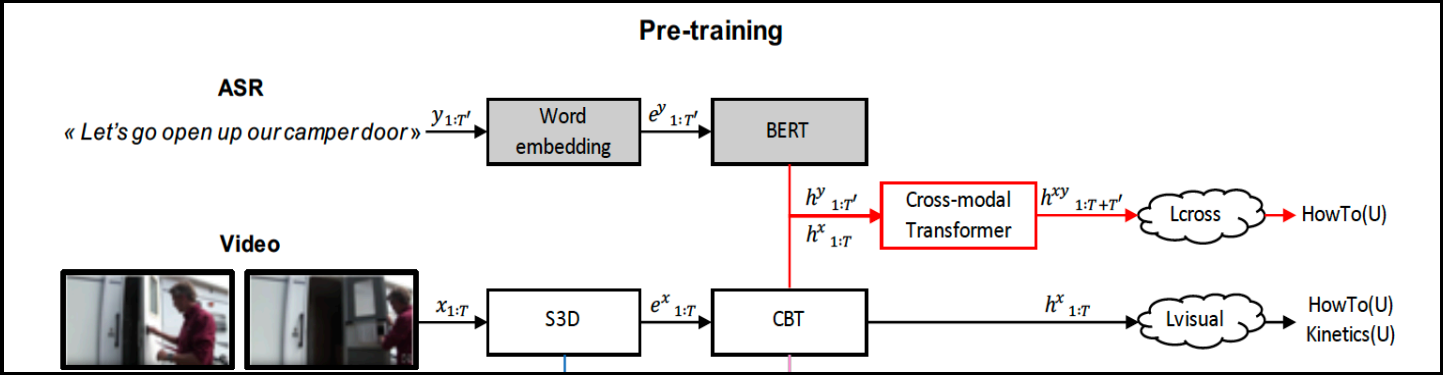
- 3 types of special token : [CLS], [>], [SEP]
- Problem : lack of semantic relatedness
- Overall, we have three training regimes : Text, Video, Video & Text

Pretrain task #1 Mask completion > Text-only, Video-only

Pretrain taks #2 Linguistic-visual alignment classification > Video-text

Train objective : Optimize the weighted sum of the individual objectives

2> LEARNING VIDEO REPRESENTATIONS USING CONTRASTIVE BIDIRECTIONAL TRANSFORMER



Objective :

Propose a way to train bidirectional transformer models on sequences of real valued vectors (e.g., video frames), using noise contrastive estimation. (removes the tokenization step)

Pre-training task : Maximizee the MI between x and y at frame-level and sequence level

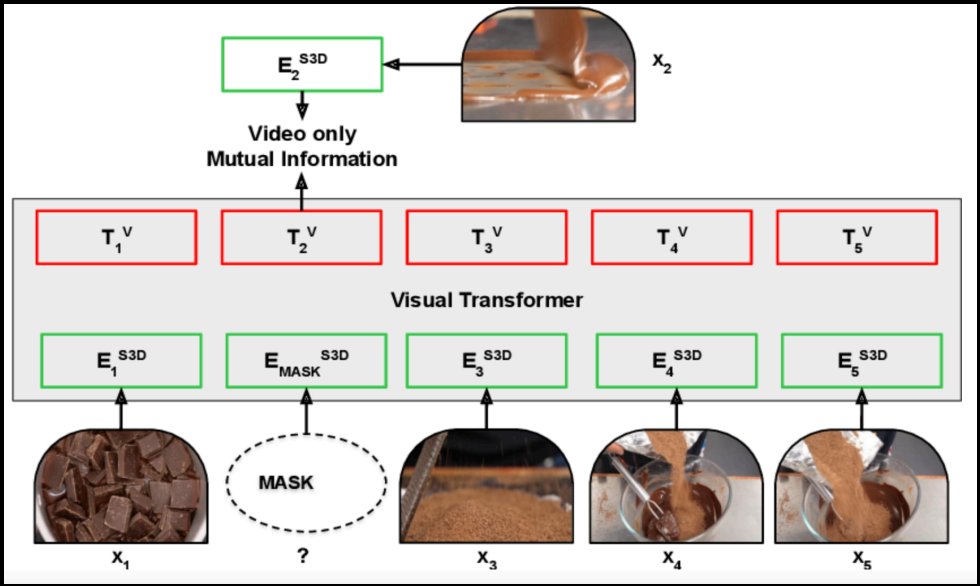
Down stream task : video classification, captioning, segmentation

Main Contribution :

- “lightweight” way of combining signals after training each modality separately.
- CBT, Cross modal transformer to maximize the mutual information video and sentence.
- Robust to small misalignments between the sequences

Method

- CBT Model applied to frames (maximize the MI between x and y at Frame-level)



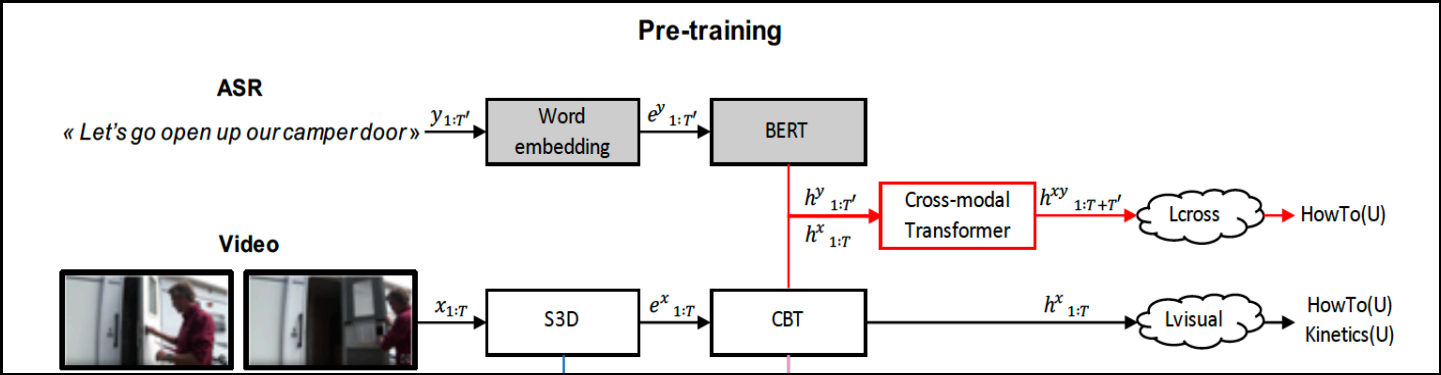
$$L_{\text{visual}} = -E_{\mathbf{x}\sim\mathcal{D}} \sum_t \log \text{NCE}(\mathbf{x}_t|\mathbf{x}_{-t})$$

$$\text{NCE}(\mathbf{x}_t|\mathbf{x}_{-t}) = \frac{\exp(\mathbf{e}_t^T \hat{\mathbf{e}}_t)}{\exp(\mathbf{e}_t^T \hat{\mathbf{e}}_t) + \sum_{j\in\text{neg}(t)} \exp(\mathbf{e}_j^T \hat{\mathbf{e}}_t)}$$

- $e_t = f_{enc}(x_t)$ is the output of a 3D CNN, $\hat{\mathbf{e}}_t = g_{\text{context}}(\mathbf{e}_{-t})$ is the output of a visual transformer,
- neg(t) is a set of negative samples (all the other frames from the same minibatch as frame t)

Method

Cross-Modal CBT Model (maximize the MI between x and y at Sequence-level)



To do this, we first encode each sequence using CBT and BERT to get $\mathbf{h}^x_{1:T} = \text{CBT}(\mathbf{x}_{1:T})$ and $\mathbf{h}^y_{1:T'} = \text{BERT}(\mathbf{y}_{1:T'})$, as shown in fig. 1. We then concatenate these sequences and pass them to a shallow cross-modal transformer to produce $\mathbf{h}^{xy}_{1:T+T'}$. Finally, we pass this to a shallow MLP to compute an MI-like score $\text{MI}(\mathbf{x}, \mathbf{y}) = f(\mathbf{h}^{xy}_{1:T+T'})$. (Here $f()$ extracts the features from $\mathbf{h}^{xy}_0$, but it could also use average pooling.) This model is trained using $L_{\text{cross}} = -E_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}} \log \text{NCE}(\mathbf{y}|\mathbf{x})$, where

$$\text{NCE}(\mathbf{y}|\mathbf{x}) = \frac{\text{MI}(\mathbf{x}, \mathbf{y})}{\text{MI}(\mathbf{x}, \mathbf{y}) + \sum_{\mathbf{y}' \in \text{Neg}(\mathbf{y})} \text{MI}(\mathbf{x}, \mathbf{y}')} \tag{5}$$

where $\text{Neg}(\mathbf{y})$ is a set of ASR sequences not associated with video $\mathbf{x}$.

Method

- Overall model
  - one transformer (BERT) that takes discrete ASR tokens, one transformer that takes continuous video features, and a third transformer to estimate mutual information between two modalities.

$$L_{\text{cbt}} = w_{\text{bert}} L_{\text{bert}} + w_{\text{visual}} L_{\text{visual}} + w_{\text{cross}} L_{\text{cross}}$$

W_{bert} = 0, since we use a pre-trained (frozen) BERT model for ASR. We set W_{visual} = 1, and either W_{cross} =1 or 0, depending on whether we use cross-modal training or not.

Related work survey <Multi-modal Self-supervision>

Method	Architecture	Visual token
VideoBERT <ICCV 19>	One single modal BERT + One single modal video token + One cross-modal transformer	Vector quantization (Video)
Contrative Bidirectional Transformer <Under Review>	One single modal BERT + One single modal CBT(Video) + One cross-modal CBT (1 or 0)	Output of a 3D CNN
ViLBERT <NeurIPS 19>	Two stream	Image RoI
InterBERT <Under Review>	Single stream & Two stream	Image RoI
Pixel-BERT <Under Review>	Single stream	Full Image

ViLBERT:
Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks

Jiasen Lu, Dhruv Batra, Devi Parikh, Stefan Lee

Georgia Institute of Technology, Oregon State University, Facebook AI Research

NeurIPS 2019

Overview

- Motivation : A model for learning task-agnostic joint representation of image content and natural language

- Method : ViL-BERT (Vision & Language for BERT)
  - Extend BERT architecture to a multi-modal two-stream model,
  - processing both visual and textual inputs in separate streams that interact through co-attentional transformer layers

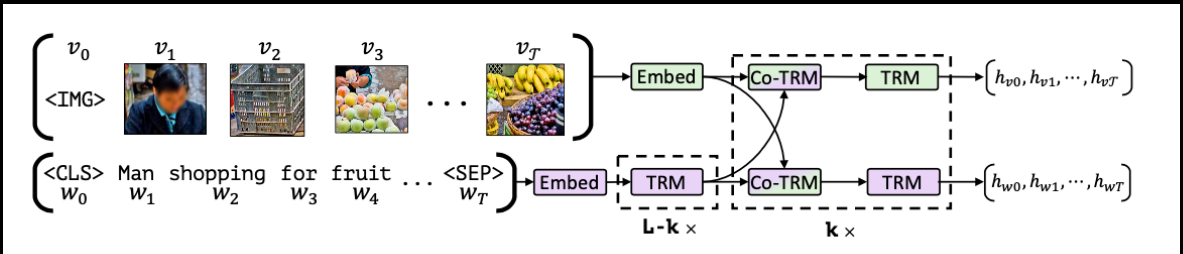
- Visual Token : Image ROI

- Pre-train Dataset : Conceptual Captions

- Pre-train Tasks : Masked Language/Object Modeling, Image-Text Matching

- Downstream Tasks : VQA, VCR, Grounding referring expressions, Image retrieval, Zero-shot image retrieval

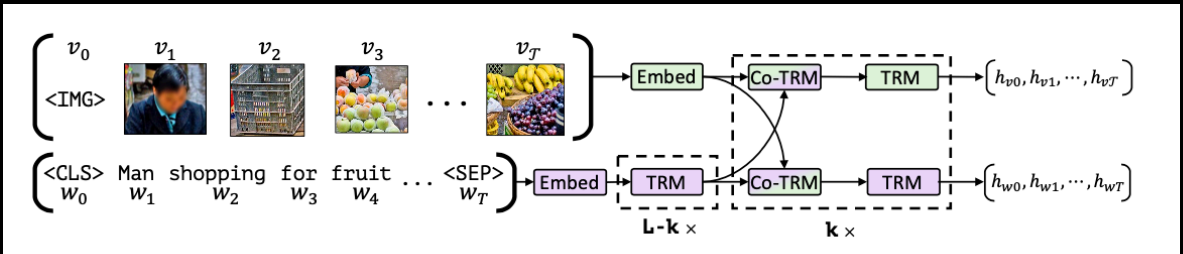
ViLBERT: Extending BERT for Jointly Represent Images and Text



Architecture of ViLBERT

- **Image representations**
  - Generate image region features by extracting bounding box and their visual features from a pre-trained Fast R-CNN
  - Encode spatial location, constructing a 5-d vector from region position and the fraction of image area covered
  - Mark the beginning of an image region sequence with a special IMG token representing the entire image

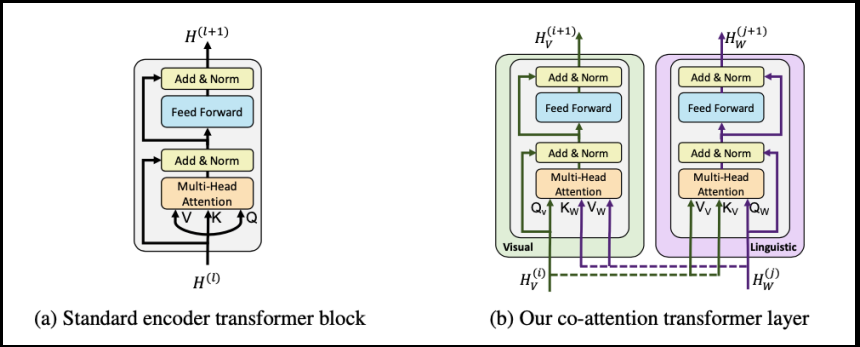
ViLBERT: Extending BERT for Jointly Represent Images and Text



Architecture of ViLBERT

- **Two-stream architecture**
  - Two parallel BERT-style models
    operating over image regions and text segments
  - Modeling each modality separately,
    then fusing them through a small set of attention-based interactions
  - Each stream is a series of transformer blocks(TRM)
    and novel co-attentional transformer layers(Co-TRM)

Co-Attentional Transformer Layers
: enable information exchange between modalities



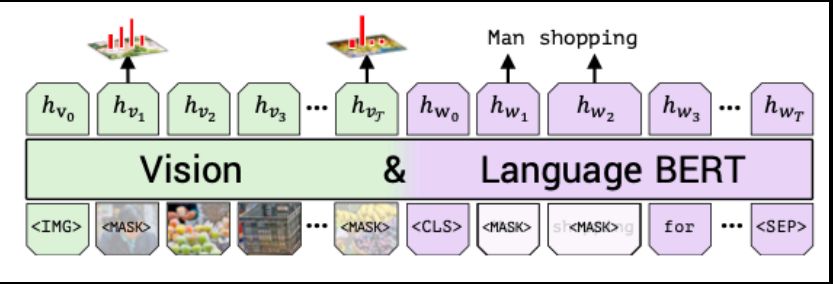
(a) Standard encoder transformer block (b) Our co-attention transformer layer

Proposed Co-Attention Transformer Layer

- **Co-TRM**
  - By exchange key-value pairs in multi-headed attention
  - Performing image-conditioned language attention in the visual stream and language-conditioned image attention in the linguistic stream
  - The rest of the transformer block proceeds as before

Training Tasks and Objectives

: train on the Conceptual Captions dataset to learn visual grounding



Masked Language/Object Modeling task

- **Masked multi-modal learning**
  - Reconstruct image region categories or words for masked input given the observed inputs
  - Rather than directly regressing the masked feature values, the model instead predicts a distribution over semantic classes for the corresponding image region
  - To supervised this, we take the output distribution for the region from the same pre-trained detection model used in feature extraction

Training Tasks and Objectives
: train on the Conceptual Captions dataset to learn visual grounding

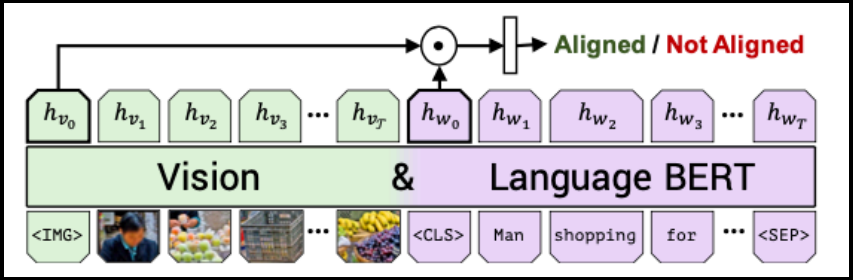


Image-Text Matching task

- **Multi-modal alignment prediction**
  - Predict whether the image and text are aligned, i.e. whether the text describes the image
  - Compute the overall representation as an element-wise product and learn a linear layer to make the binary prediction whether the image and text are aligned

InterBERT: An Effective Multi-Modal Pretraining Approach via Vision-and-Language Interaction

Juyang Lin, An Yang, Yichang Zhang, Jie Liu, Jingre Zhou, Hongzia Yang

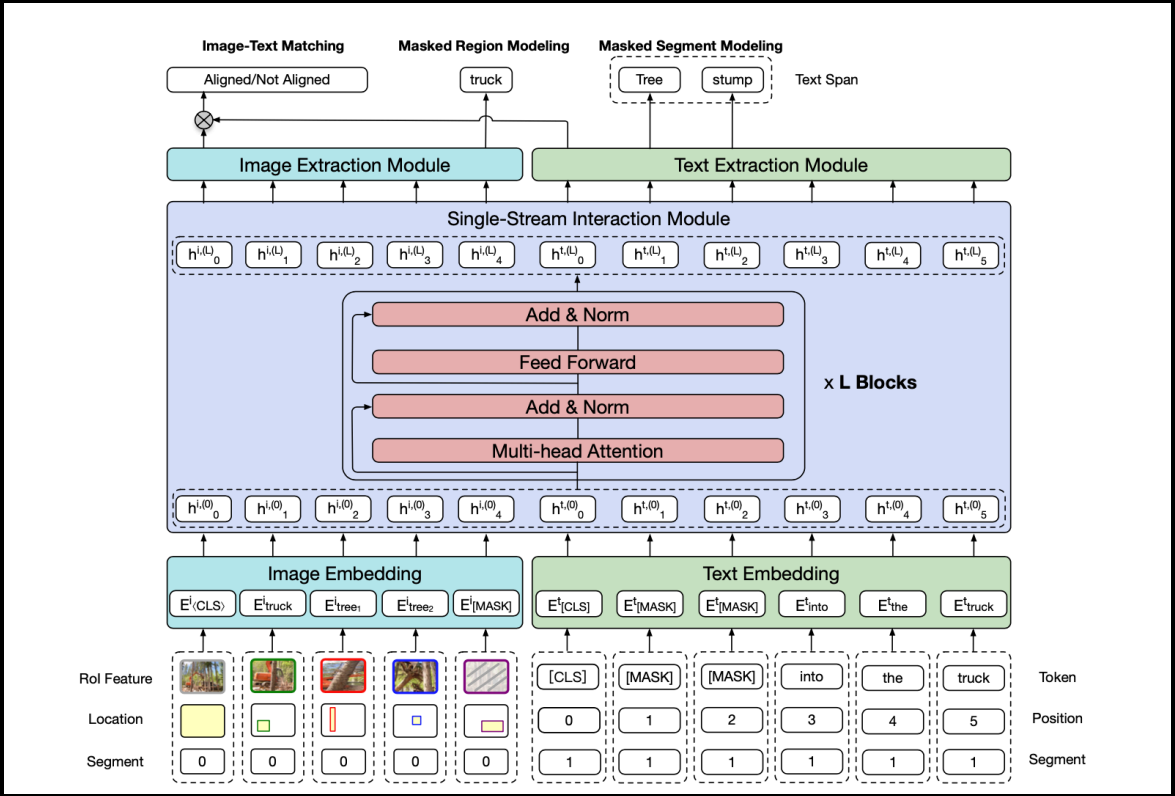
Alibaba Group

Under review

Overview

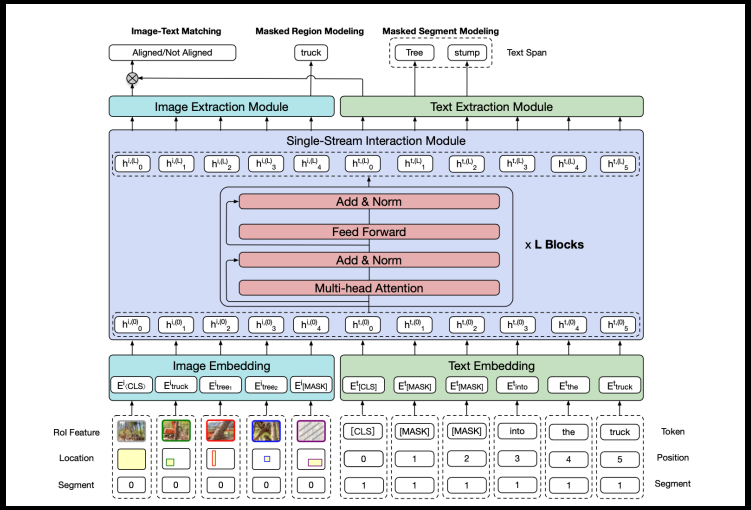
- Motivation : Modeling interaction between the information flows of different modalities
- Method : InterBERT (BERT for Interaction)
  - Consists of a single-stream interaction and a two-stream extraction module
- Visual Token : Image ROI
- Pre-train Dataset : Conceptual Captions, SBU Captions, COCO captions
- Pre-train Tasks : Masked Segment/Region Modeling, Image-Text Matching
- Downstream Tasks : VCR, Image retrieval, Zero-shot image retrieval

InterBERT : Interaction for BERT



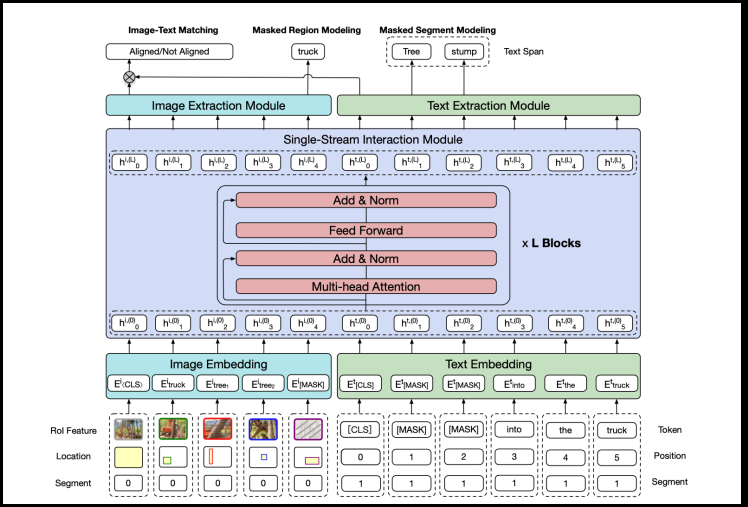
Architecture of InterBERT

Model architectures



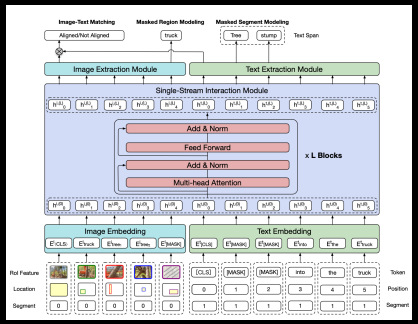
- **Single-Stream Interaction Module**
  - Replacing co-attention with all-attention
    - co-attention ignore the self-context
  - This architecture enables strong interaction between modalities with the attention mechanism
    - a combination of self attention and co-attention
    - the model can generate more contextualized representations

Model architectures



- Two-Stream Extraction Module
  - Develop a module to respectively generate representations to separate the fused information
  - Preserving modal independence
- Image embedding
  - Obtain the object representations and their locations with a detector
    - Use object detector Faster-RCNN trained on Visual Genome
    - Extract the bounding boxes and the ROI features
  - Apply feature, positional and segment embedding to the extracted features

Pre-training tasks



- **Masked group modeling**
  - **To predict the masked words and the categories of the masked objects**
  - **Masked segment modeling (MSM) on text**
    - **masks a continuous segment of text instead of random words**
  - **Masked region modeling (MRM) on image**

For MSM, we randomly choose words as masking anchors by the probability of 10%, and we randomly mask the anchors and 0 to 2 words after the anchors by the probability of uniform distribution. For MRM, we also randomly choose objects as masking anchors by the probability of 10%, and we mask the objects whose IoUs with the anchors are larger than 0.4. The objective of the model is to predict the masked words and the categories of the masked objects. The training minimizes the loss:

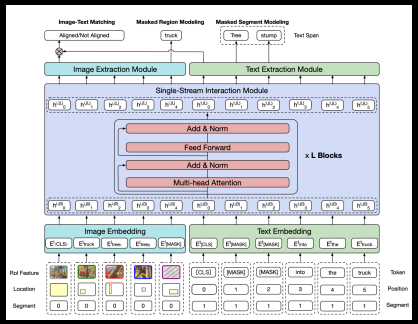
$$\mathcal{L}_{\mathbf{MSM}} = -\mathbb{E}_{x\sim D} \log p\left(\overline{x}|\hat{x}\right) \approx -\frac{1}{N}\sum_{n=1}^{N}\sum_{t=1}^{T} \mathbf{m_t}(x^n, t) \log p_\theta\left(x_t^n|\hat{x}^n\right), \tag{5}$$

$$\mathcal{L}_{\mathbf{MRM}} = -\mathbb{E}_{x\sim D} \log p\left(\overline{x}|\hat{x}\right) \approx -\frac{1}{N}\sum_{n=1}^{N}\sum_{t=1}^{T} \mathbf{m_i}(x^n, t) \log p_\theta\left(x_t^n|\hat{x}^n\right), \tag{6}$$

where x is a random sample of image-text pair from the training set D, and $\overline{x}$ refers to the masked segment or the masked region, and $\hat{x}$ refers the whole masked sequence x. $\mathbf{m_i}$ and $\mathbf{m_t}$ refer to the masking functions for image and text. The objective functions encourage the model to predict the masked groups of words or the class of the masked groups of objects.

Loss functions of MSM, MRM

Pre-training tasks



- **Image-text matching**
  - **For learning the relation between image and text, whether the image and text are align**

$$\mathcal{L}_{\mathbf{ITM}} = -\mathbb{E}_{x,y\sim D}\left[y \log p\left(y|\hat{x}\right) + (1-y)\log\left(1-p\left(y|\hat{x}\right)\right)\right], \tag{7}$$

where x is a random sample from the training set D and $y \in \{0,1\}$ denotes whether x is positive or negative. $\hat{x}$ refers to the masked x.

The overall objective function is the weighted sum of the aforementioned terms, as shown below:

$$\mathcal{L} = \lambda_1 \mathcal{L}_{\mathbf{MSM}} + \lambda_2 \mathcal{L}_{\mathbf{MRM}} + \lambda_3 \mathcal{L}_{\mathbf{ITM}}, \tag{8}$$

where λ refers to the hyperparameter for the weights for each term.

Overall loss function

Pixel-BERT: Aligning Image Pixels with Text by Deep Multi-Modal Transformers

Zhicheng Huang, Zhaoyang Zeng, Bei Liu, Dongmei Fu, and Jianlong Fu

University of Science and Technology Beijing, Sun Yat-sen University, Microsoft Research

arXiv

Overview

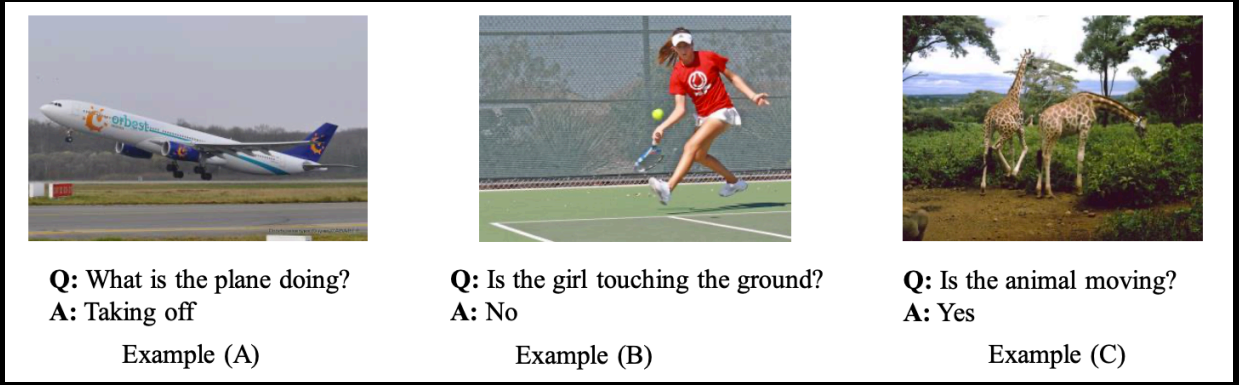
- Motivation : Jointly learn visual and language embedding
  in a unified end-to-end framework
- Method : Pixel-BERT, Random pixel sampling mechanism
- Visual Token : Full image
- Pre-train Dataset : Visual Genome, MS-COCO
- Pre-train Tasks : Masked Language Modeling, Image-Text Matching
- Downstream Tasks : VQA, Image retrieval,
  Natural Language for Visual Reasoning for Real

Motivation

difficult to obtain the status of the plane

hard to judge the actual spatial relation between “girl” and “ground”

hard to infer the status of the animals

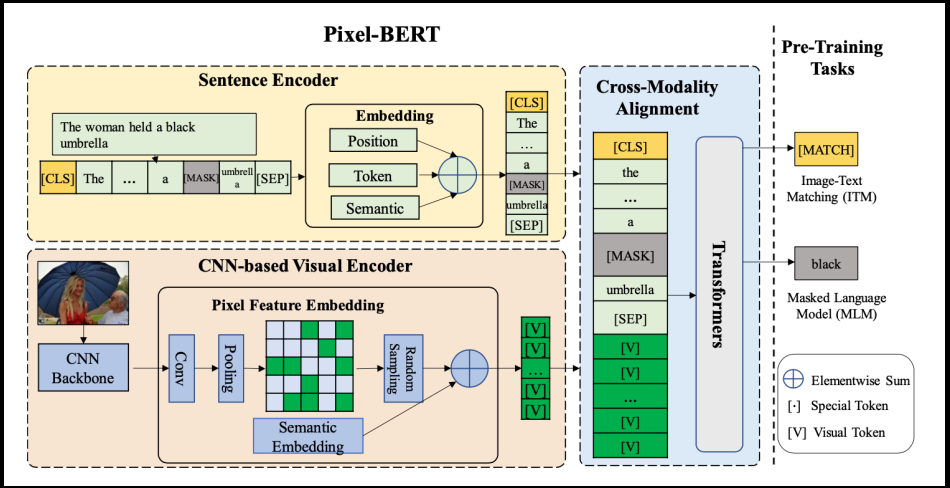


Examples of image, questions and answers in VQA2.0 dataset

- Region-based visual features cannot well handle
  - feature extractors are designed for specific visual tasks

    e.g. object detection
  - this will cause an information gap with language understanding
  - Some important factors of visual information are lost
  - Visual information of much broader semantics are lost

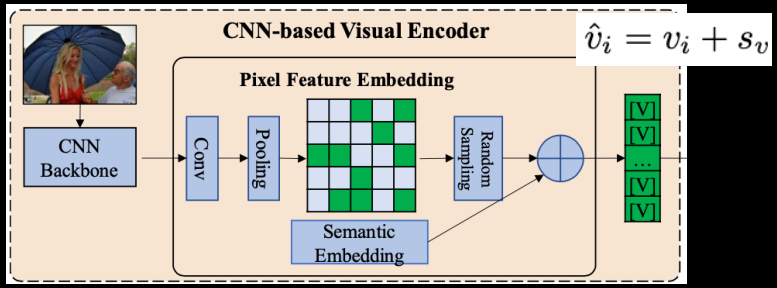
Pixel-BERT



Architecture of Pixel-BERT

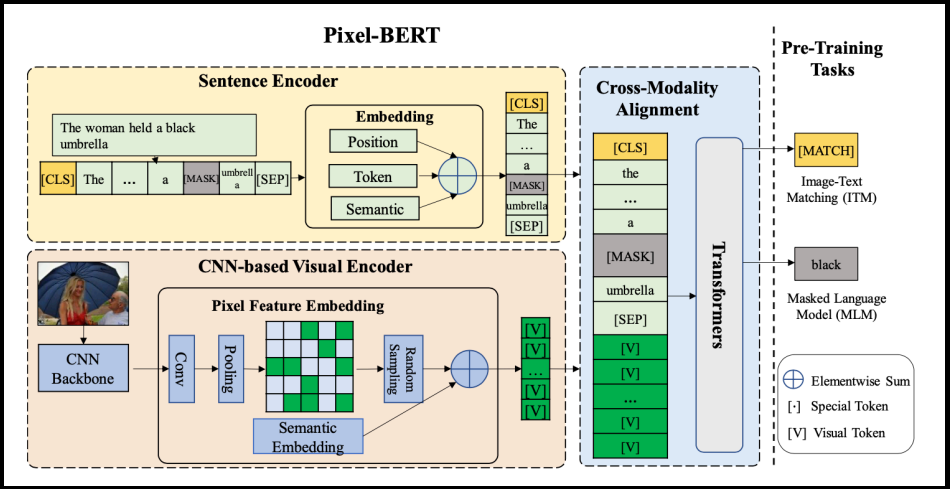
- To fully utilize visual information of the original images
- Learning visual embedding from pixels, named Pixel-BERT
  - CNN-based Visual Encoder
  - Sentence Encoder, based on BERT
  - Cross-Modality Alignment

CNN-based Visual Encoder



- **Image Feature Embedding**
  - Given an input image I, use CNN backbone to extract its feature
  - then flat the feature along the spatial dimension
    - $V = \{v_1, v_2, \ldots, v_k\} \in R^d$, k indicates the number of feature pixels
- **Random Sampling**
  - randomly sample a fixed number of 100 pixels from the feature map
  - To improve the robustness of feature learning and avoid overfitting

Cross-Modality Module



- **Adopt Transformer to learn cross-modality attention between image pixels and language tokens**
  - **Combine all vectors to construct the input sequence**
  - **Also adding tow special tokens [CLS], [SEP]**
  - **The final input sequence to the join-learning Transformer is**

$$\{[\mathrm{CLS}], \hat{w}_1, \hat{w}_2, \cdots, \hat{w}_n, [\mathrm{SEP}], \hat{v}_1, \hat{v}_2, \cdots, \hat{v}_k\}.$$

Result

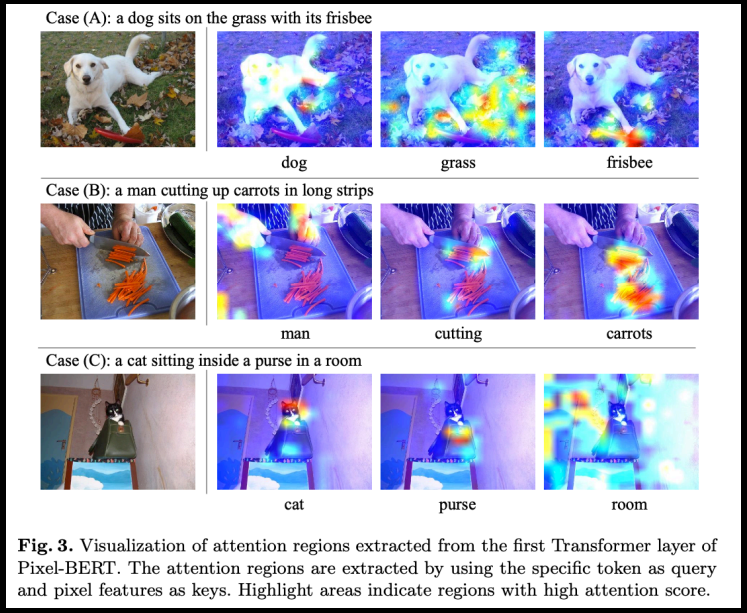


Fig. 3. Visualization of attention regions extracted from the first Transformer layer of Pixel-BERT. The attention regions are extracted by using the specific token as query and pixel features as keys. Highlight areas indicate regions with high attention score.

Visualization of attention regions of Pixel-BERT

- **Although we did not apply any spatial supervision to guide the attention learning, the results from Fig.3 show that with well-define tasks**

Our Strategy - Objective

1. Self-supervised image-text embedding model.
2. Image classification and generating radiology text.
3. Classification result AUC.
4. Highlight the ROI by heatmap.

Our Strategy - Method

Multi-modality (Free text & Image)

1> Radiology report generation (Image captioning)

- CNN + RNN model
- Automatic Generation of Medical Reports
- Evaluate NLG result with other NLP Tool to measure accuracy, precision, F1 score.

2> Lesion detection and annotation

- CNN + RNN model
- Lesion detection with highlight the ROI by heatmap
- AUC result and IOU result with NIH-dataset

3> 1 + 2

- CNN + RNN model
- Report Generation
- Lesion and radiology activation map visualization
- AUC result and Model evaluation with labeler NLP tools

Self-supervision

1> Contrastive learning for Image representation

& BERT for Natural Language

2> BERT based self-supervision (Image & Natural Language)

Our Strategy – scenario 1