Describing Visual Stimuli Using Learnt Semantic Representations of Brain Activity

Team Tumornators

Kushagra Agarwal, Shantanu Agrawal, Shreeya Pahune

Background

Many neuroscience studies have attempted to quantitatively analyze the semantic representation of what a human recalls using the fMRI data of brain activity evoked by visual stimuli, such as natural movies and images. These try to assign semantic labels to still pictures using natural language descriptions synchronized with the pictures and discuss the relationships between the visual stimuli evoked by the still pictures and brain activity. Based on these relationships, they construct a model to classify brain activity into semantic categories to reveal areas of the brain that deal with particular semantic categories. We in this project will try to extract these learnt semantic representations and attempt to generate the corresponding natural language descriptions for the brain data.

Problem Definition

The problem statement we are trying to address is to generate natural language descriptions using brain activity data as input. We first learn image representations from fMRI data and then use these learnt representations to generate image captions.

The project broadly aims to generate natural language sentences that describe what a human being calls to mind using brain activity data observed by fMRI as input information (Matsuo et al., 2018).

Overview of the approach

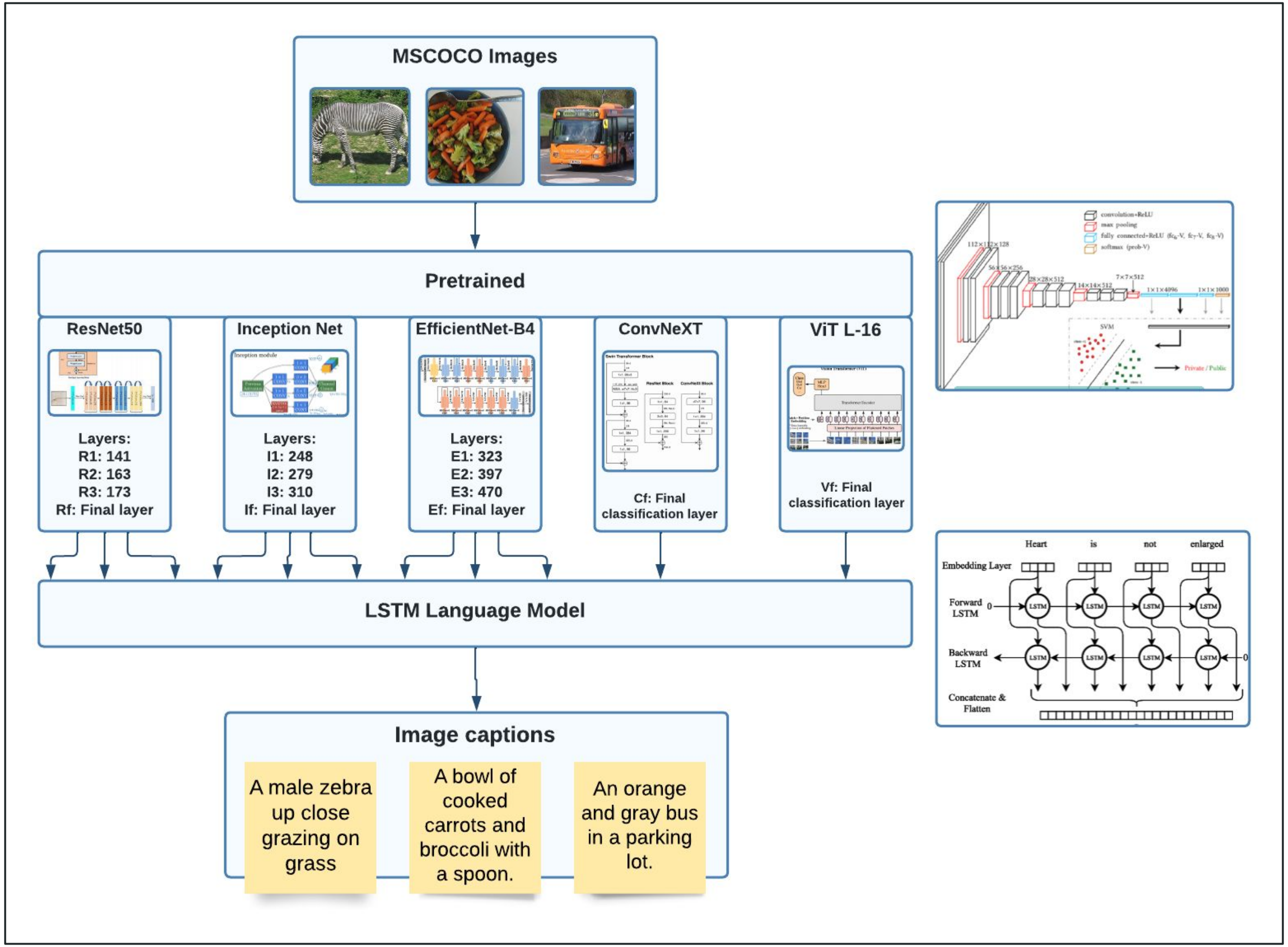
- First, we trained an image captioning model on MSCOCO images and their corresponding captions. The image captioning model consists of a pretrained Convolutional Neural Network to generate image features from an input image and an LSTM model to predict captions using these image features.
- Next, we used fMRI images obtained on visual stimuli to generate the image features. This was done by extracting image features for the original image using the CNN model and then training an ML model to predict these image features.
- Finally, the two modules were combined, i.e, Brain fMRI -> Image Features and Image Features -> Captions to finally get the desired fMRI->Captions output.

Pipeline 1 (Image -> Captions)

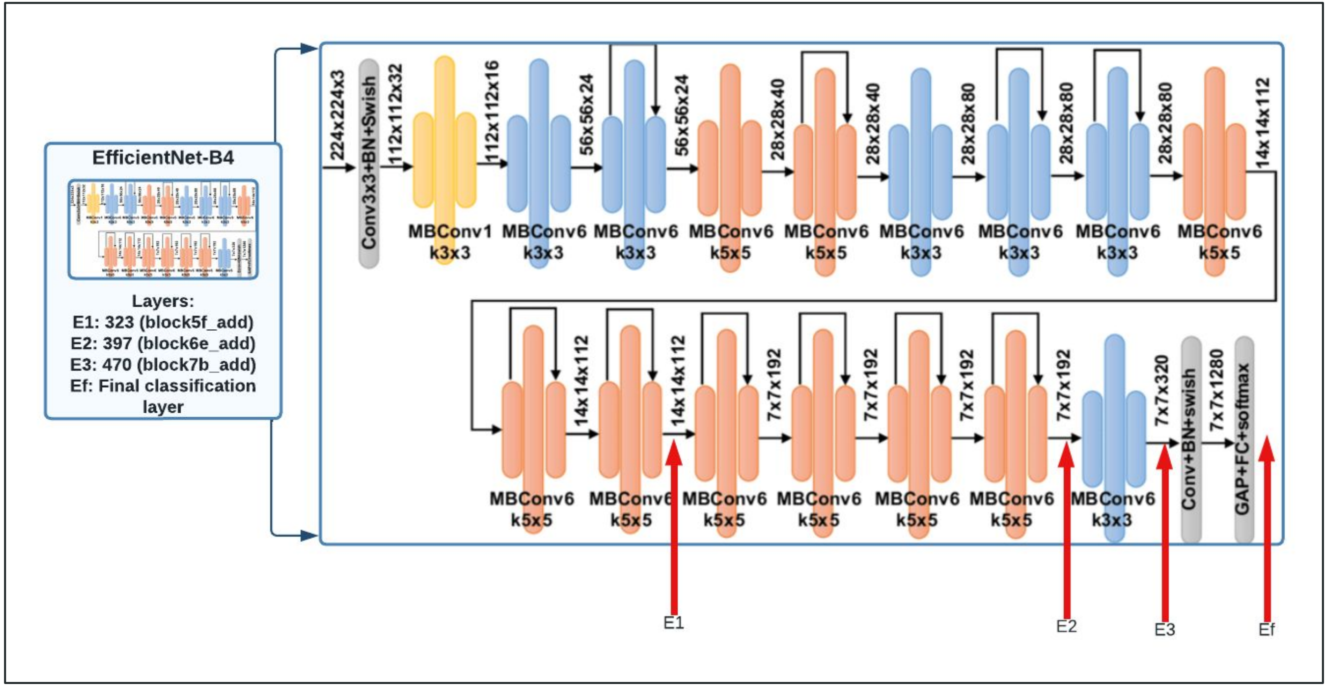
Our first pipeline consists of two parts:

A. **Encoder**
   Image feature embeddings are obtained for all input MSCOCO images from five pretrained models (ResNet50, Inception Net, Efficient Net-B4, ConvNeXT and ViT L-16). For each of these models, embeddings from 4 layers were extracted (3 intermediate layers and 1 final fully connected layer).

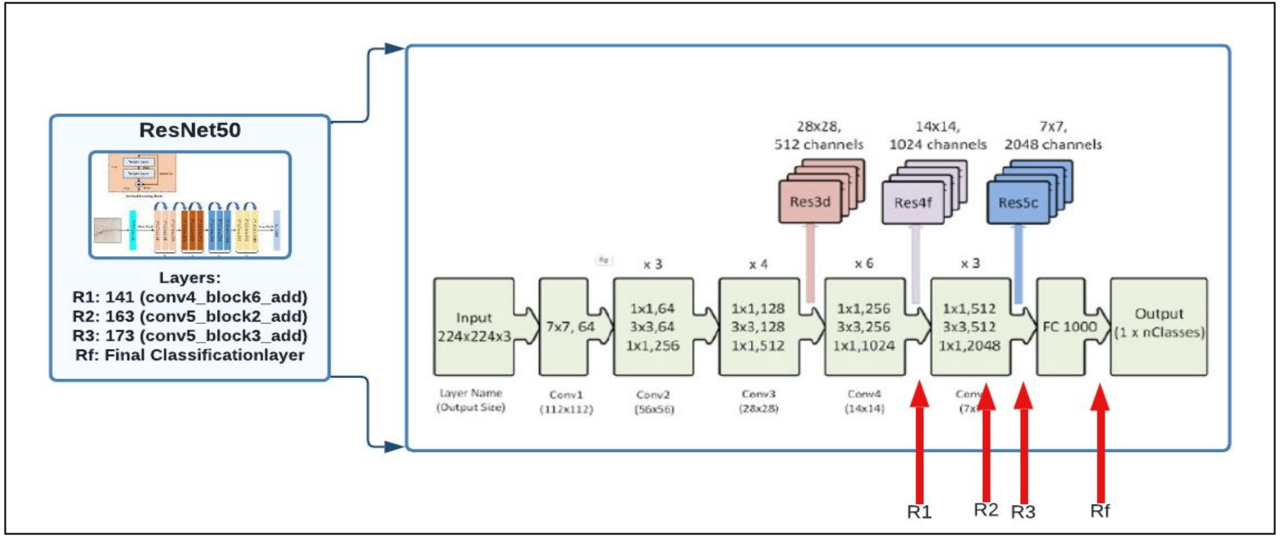
B. **Decoder**
   With input as these image features, we train a LSTM for each layer of each model with corresponding image captions. The output is evaluated using BLEU scores.


MSCOCO Images
Pretrained
ResNet50
Layers:
R1: 141
R2: 163
R3: 173
Rf: Final layer
Inception Net
Layers:
I1: 248
I2: 279
I3: 310
If: Final layer
EfficientNet-B4
Layers:
E1: 323
E2: 397
E3: 470
Ef: Final layer
ConvNeXT
Cf: Final classification layer
ViT L-16
Vf: Final classification layer
LSTM Language Model
Image captions
A male zebra up close grazing on grass
A bowl of cooked carrots and broccoli with a spoon.
An orange and gray bus in a parking lot.
Heart
is
not
enlarged
Embedding Layer
Forward LSTM
Backward LSTM
Concatenate & Flatten

Model Architectures

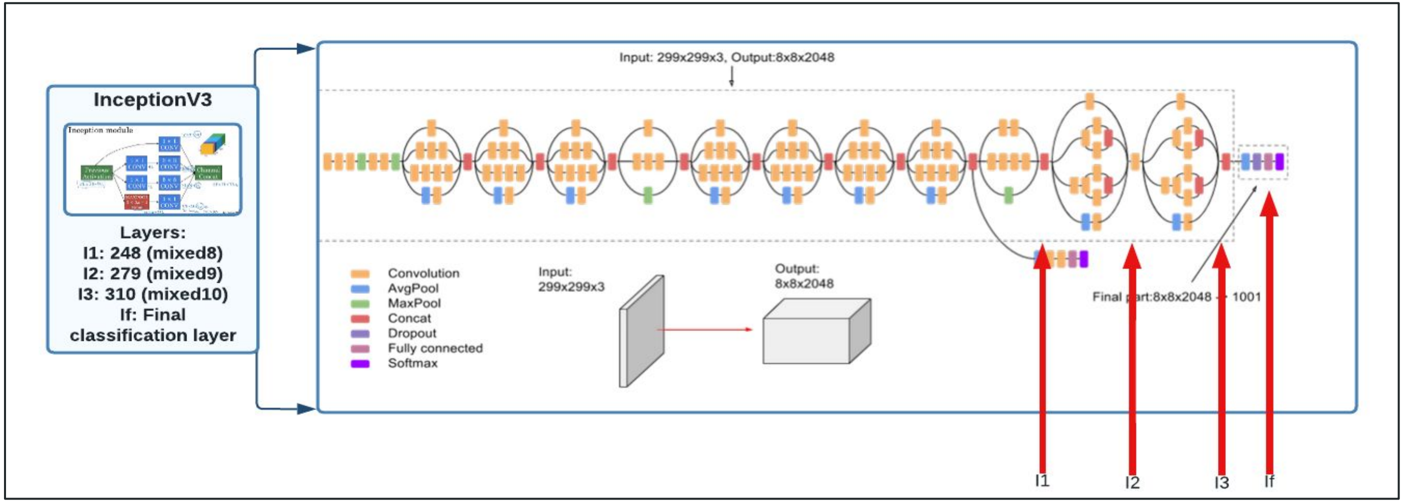


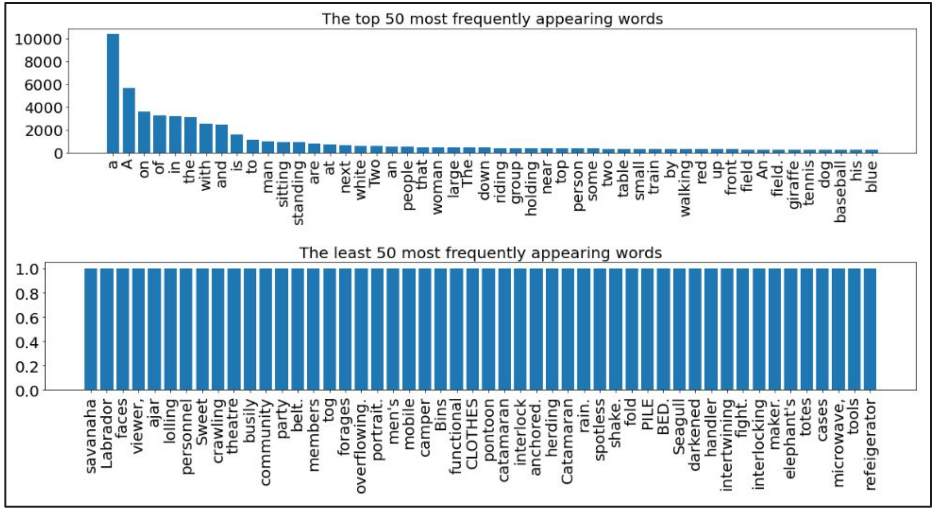
EfficientNet - B4



ResNet50

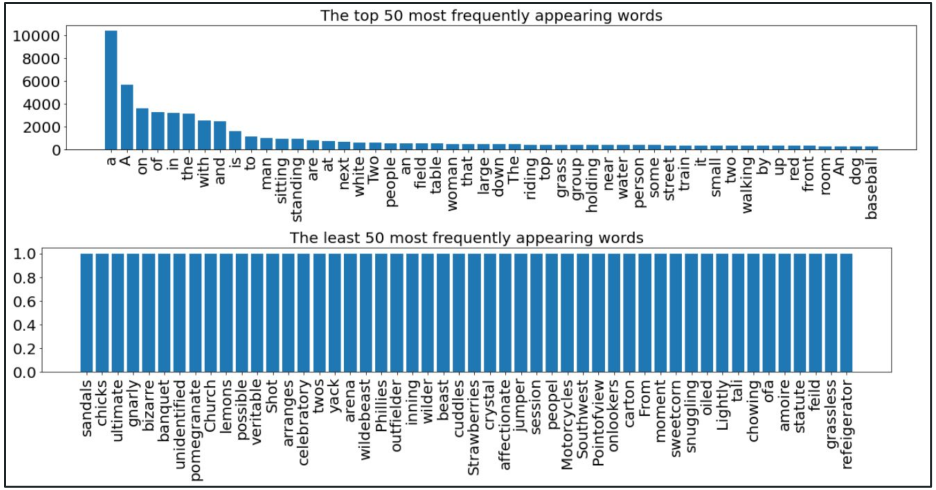
InceptionNet V3





Caption preprocessing

Removed numeric, alphanumeric, punctuations.



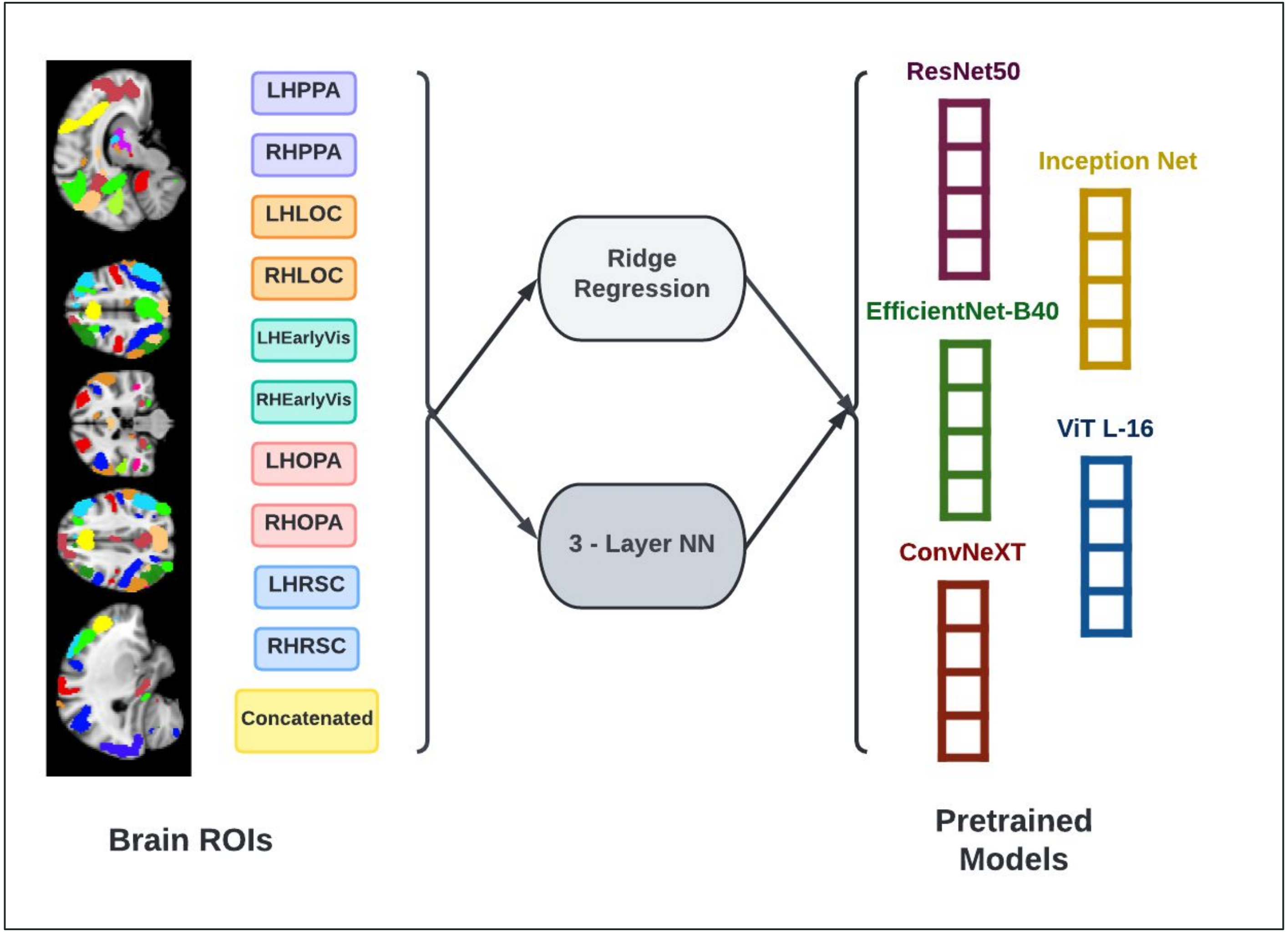
```
 Layer (type)                   Output Shape         Param #     Connected to
==================================================================================================
 input_16 (InputLayer)          [(None, 32)]         0           []

 embedding_7 (Embedding)        (None, 32, 64)       122432      ['input_16[0][0]']

 input_15 (InputLayer)          [(None, 1024)]       0           []

 CaptionFeature (LSTM)          (None, 256)          328704      ['embedding_7[0][0]']

 ImageFeature (Dense)           (None, 256)          262400      ['input_15[0][0]']

 add_7 (Add)                    (None, 256)          0           ['CaptionFeature[0][0]',
                                                                  'ImageFeature[0][0]']

 dense_14 (Dense)               (None, 256)          65792       ['add_7[0][0]']

 dense_15 (Dense)               (None, 1913)         491641      ['dense_14[0][0]']

==================================================================================================
Total params: 1,270,969
Trainable params: 1,270,969
Non-trainable params: 0
```

Pipeline 2 (fMRI -> Image Features)

The second pipeline essentially learns to replace Image Features obtained from Pipeline 1 part (a), Encoder with learnt image representations from fMRI data. To do this, we used Ridge Regression and a 3 layer NN to perform brain decoding.

We used 10 brain ROIs namely: *'LHPPA', 'RHPPA', 'LHLOC', 'RHLOC', 'LHEarlyVis', 'RHEarlyVis', 'LHOPA', 'RHOPA', 'LHRSC',* and *'RHRSC'*. We then performed brain decoding in two ways:

a) Using each **individual ROI**, we predicted the image feature at each layer for each pretrained model. This did not give very good results as individual brain ROIs had very few voxels (~200) and predicting a 4000 length image vector from it wasn't possible due to constrained information available.
b) We then **concatenated** all the **ROIs** to form a large brain fMRI vector, using which we tried to predict the image features (brain decoding).


LHPPA
RHPPA
LHLOC
RHLOC
LHEarlyVis
RHEarlyVis
LHOPA
RHOPA
LHRSC
RHRSC
Concatenated
Ridge Regression
3 - Layer NN
ResNet50
Inception Net
EfficientNet-B40
ViT L-16
ConvNeXT
Brain ROIs
Pretrained Models

Individual ROIs

```
Ridge block1 LHPPA
PA: 0.499 , MAE: 0.023 , MSE: 0.001
Stats: Train-> MAE: 0.023 MSE: 0.001
```

```
Ridge block3 LHPPA
PA: 0.582 , MAE: 0.017 , MSE: 0.001
Stats: Train-> MAE: 0.017 MSE: 0.001
```

```
Ridge block4 LHRSC
PA: 0.867 , MAE: 0.307 , MSE: 0.191
Stats: Train-> MAE: 0.305 MSE: 0.189
```

Concatenated ROIS

```
Ridge block1
PA: 0.4977 , MAE: 0.021 , MSE: 0.0009
Stats: Train-> MAE: 0.0207 MSE: 0.0009
```

```
Ridge block3
PA: 0.5904 , MAE: 0.0157 , MSE: 0.0005
Stats: Train-> MAE: 0.0155 MSE: 0.0005
```

```
Ridge block4
PA: 0.8788 , MAE: 0.2864 , MSE: 0.1663
Stats: Train-> MAE: 0.2827 MSE: 0.1619
```

Integrated Pipeline (fMRI -> Image Features -> Captions)

Implemented end-to-end for ResNet features:

- Trained 11 Pipeline 2’s (10 for ROIs, 1 for concatenated ROIs)
- Trained 3 Layers to Caption LSTMs.

Average BLEU score: 0.54

ACTUAL Captions:

1. A woman walking down a street while talking on a cell phone.
2. A girl pulling a suitcase while walking and talking on her phone
3. A young woman on a cellphone smiling and walking.
4. A woman talking on her cell phone while walking.
5. The woman is talking on the phone while walking

PREDICTED by Pretrained Image Feature:

A woman is standing in the grass

PREDICTED by fMRI:

A man is in the grass

Cognitive Insights

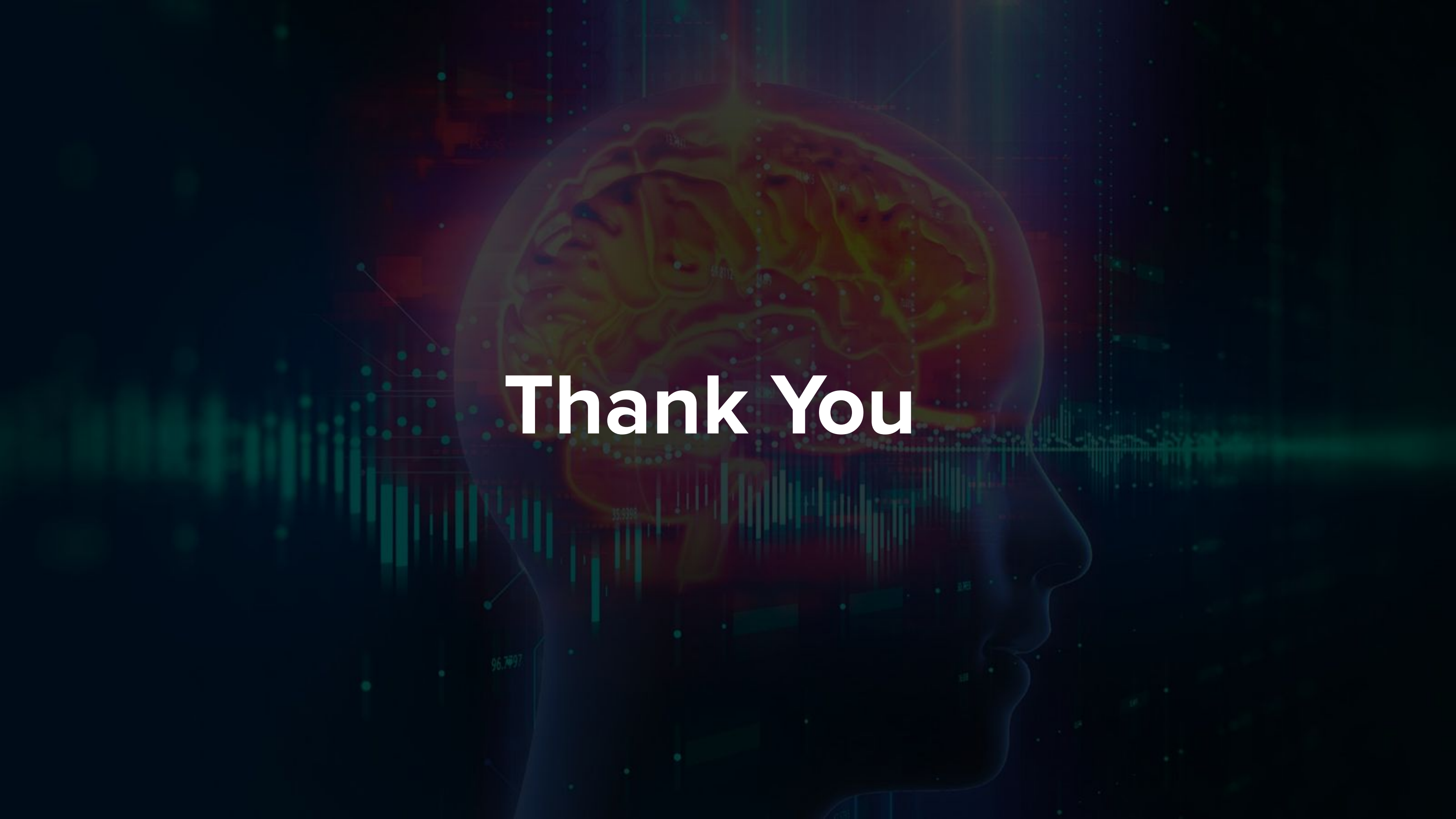
We expect that the initial layers of the pretrained network will be able to better capture edges/shapes, whereas the later layers will be able to model entire objects. This is a known property of Convolutional Neural Networks. Similarly, not all the regions of the brain perceive visual stimuli in the same manner. Some ROIs would be able to better predict earlier layers and some would be more robust to the later ones. Such distinctions can help us understand the underpinning relational structure between cognition and machine learning.

In accordance with the above hypothesis, we observe that the ROIs in general are able to better model earlier layer's (MSE:0.001) compared to later ones (MSE: 0.03). However, the increase in prediction comes at a cost of lower pairwise accuracy. This implies that the latent space of earlier layer features is more clustered than the later ones. Certain ROIs were observed to have higher accuracy in predicting earlier layers compared to others, thus validating our hypothesis.

Further Work

- We have implemented Pipeline 1 (a) for all the 5 models. PIpeline 2 has been trained over all the networks. Currently we have implemented Pipeline 1 (b) for the 4 layers of ResNet 50. Therefore, our end-to-end pipeline is currently trained on ResNet50. We plan to complete the training using the features from the remaining 4 models as well.

- Next step is to generate natural language descriptions for the brain data in Hindi as well to understand language in-specificity. Given that there exists a near perfect English->Hindi translation mapping and our fMRI->English model works, then transitively we should be able to generate Hindi descriptions using fMRIs of “non-Hindi” speaking people as well. This can be a major advance in proving the language in-specificity of visual stimuli on brain activity data.


Thank You