Attention

Discussion highlights of Tuesday 9/11/2021

Typical attention operation:

- Here we start by $qkv = \text{concat}(Q, K, V)$ all together.
- Keys, Queries, Values are $K = XW^K$, $Q = XW^Q$, $V = XW^V$
- For the attention heatmap $\mathcal{A}(K,Q) = \text{softmax}\left(QK^T / \sqrt{d_K}\right)$
- Then attention scores are $\text{Attention}(K,Q,V) = \mathcal{A}(K,Q) \cdot V$

$$X \in \mathbb{R}^{1,197,768}$$

$$qkv \in \mathbb{R}^{1,197,3\times768} = \mathbb{R}^{1,197,2304}$$

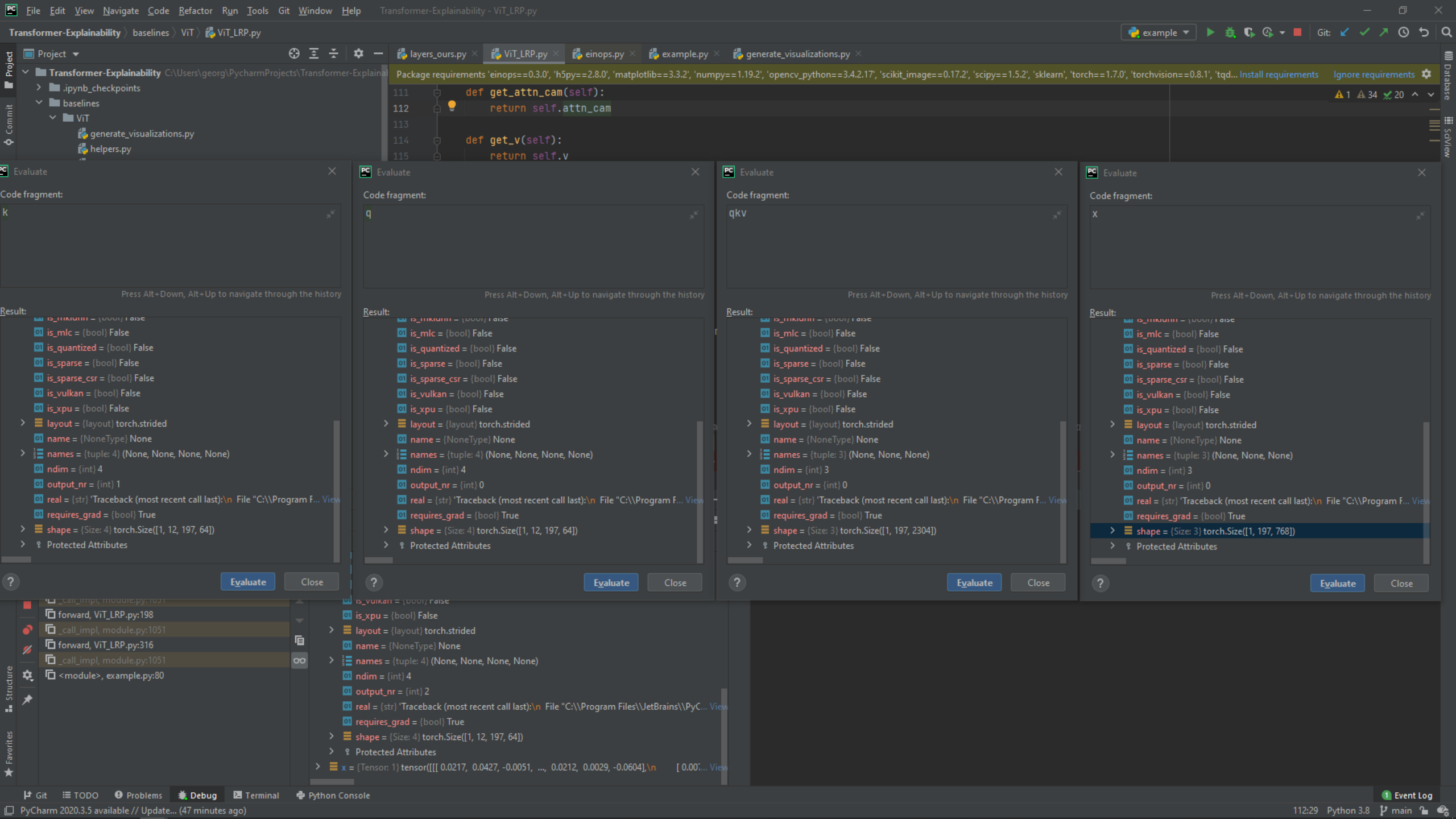
$$q \in \mathbb{R}^{1,12,197,64}$$

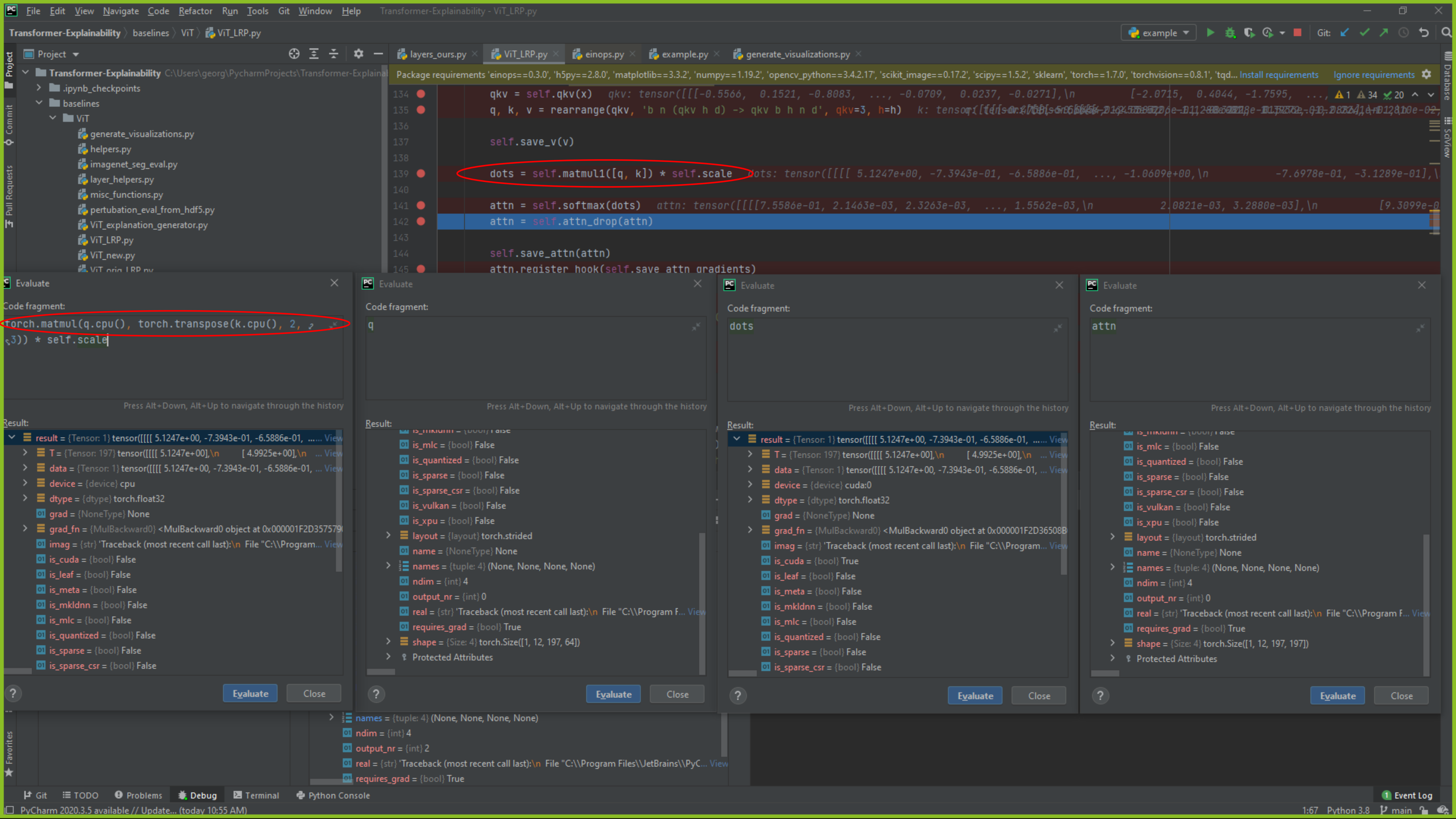
$$k \in \mathbb{R}^{1,12,197,64}$$

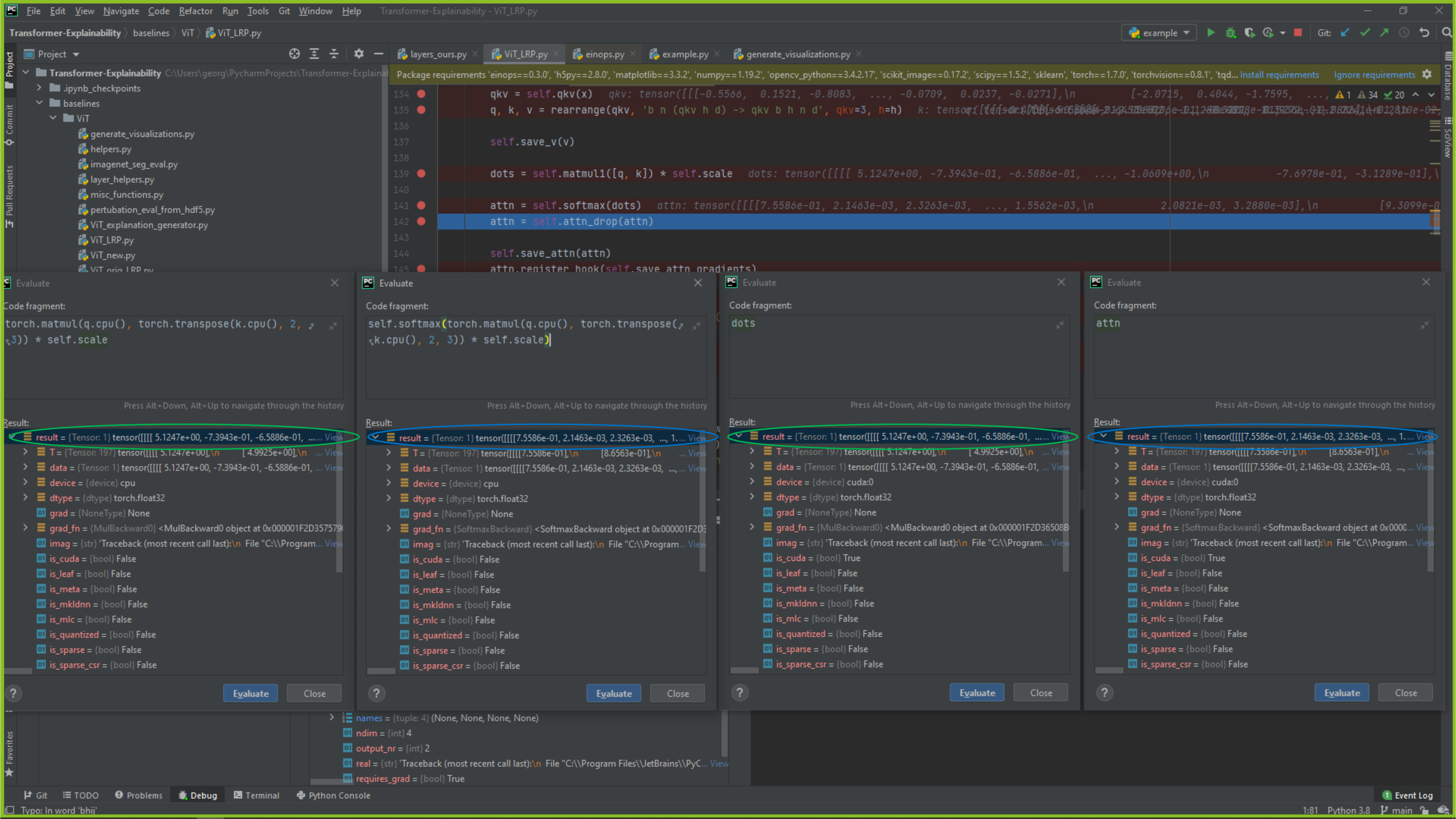
$$v \in \mathbb{R}^{1,12,197,64}$$

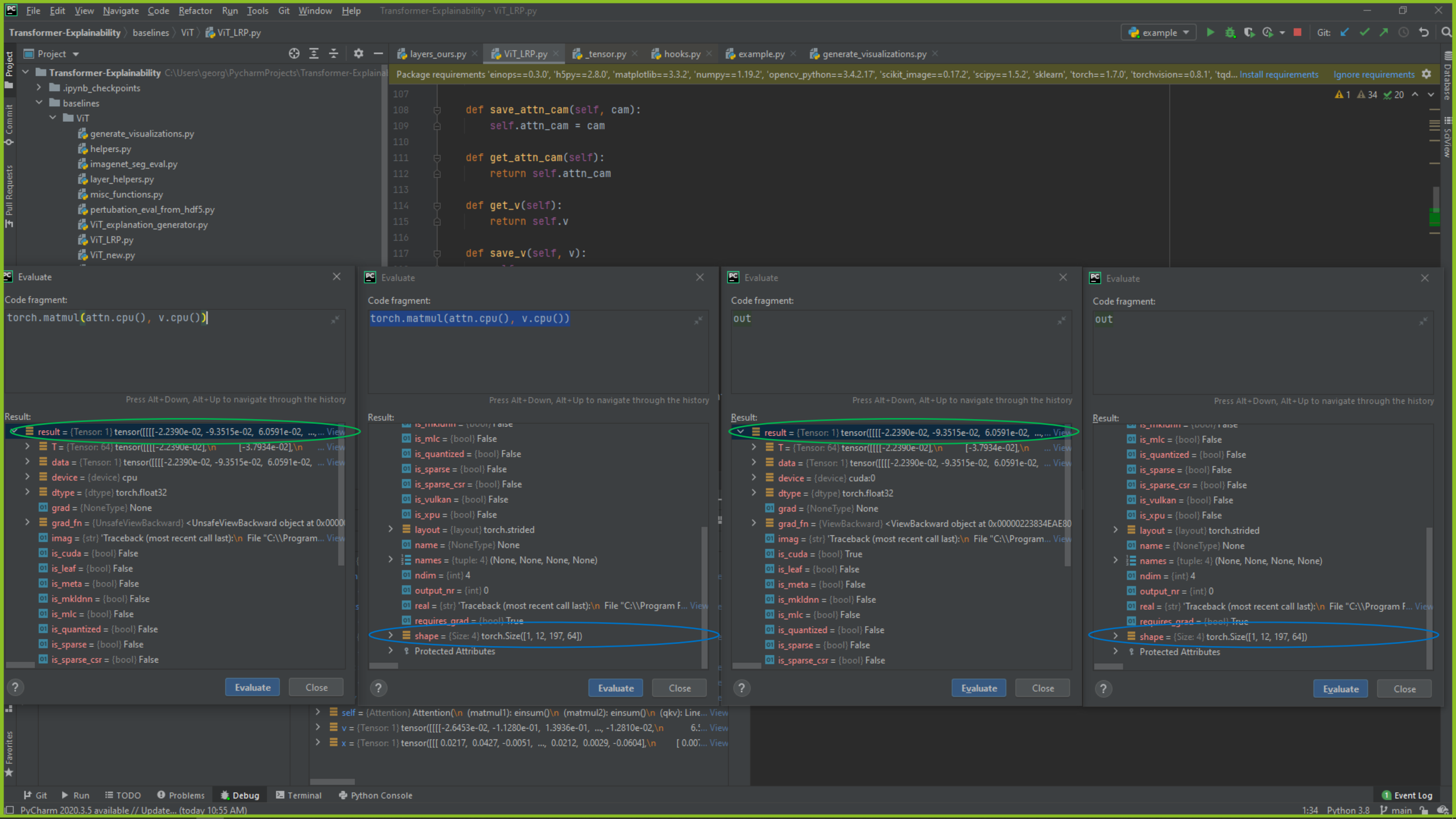
$$768 = 12 \times 64$$

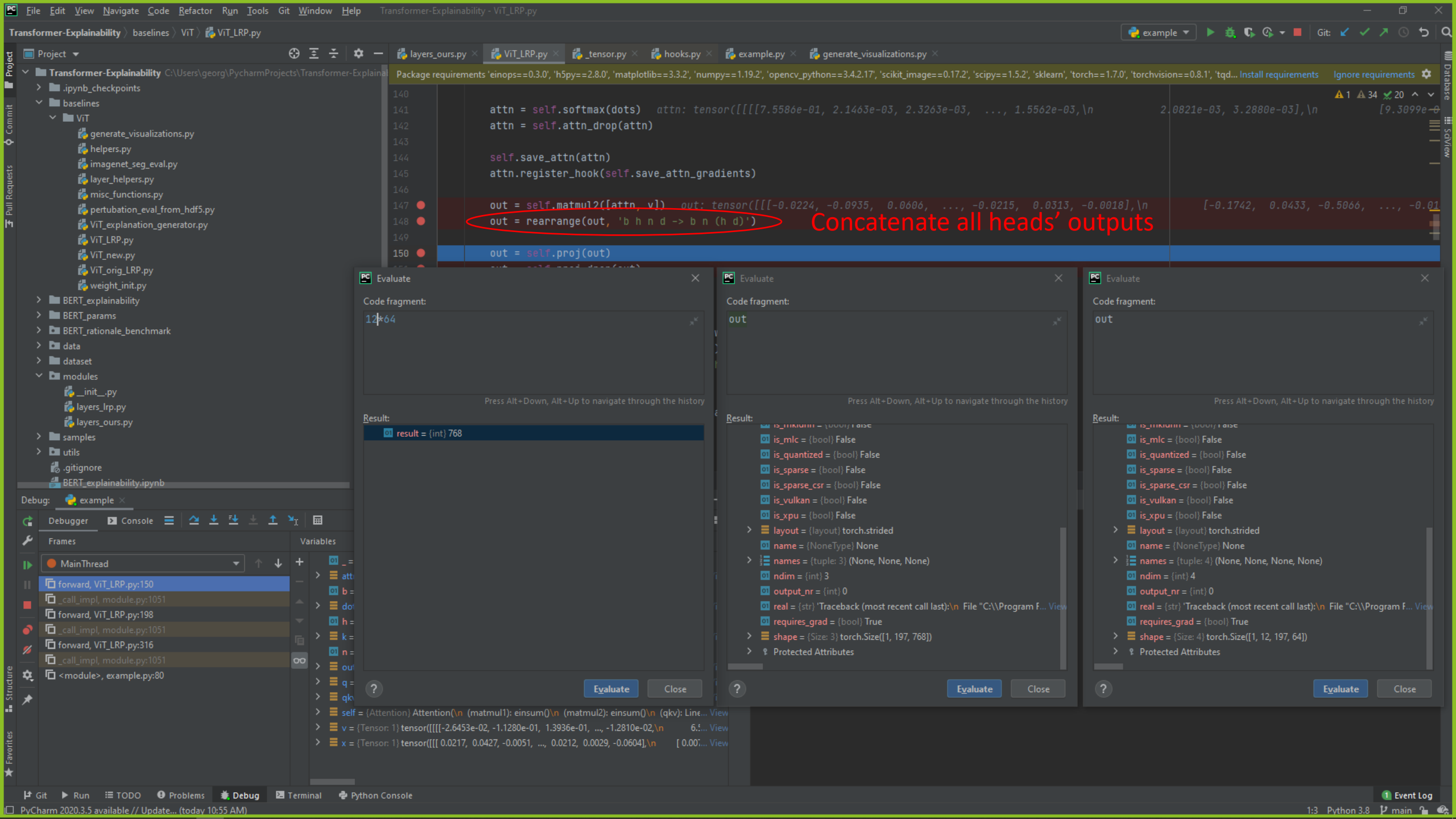
$$\sqrt{d_K} = \sqrt{\frac{768}{12}} = 0.125$$

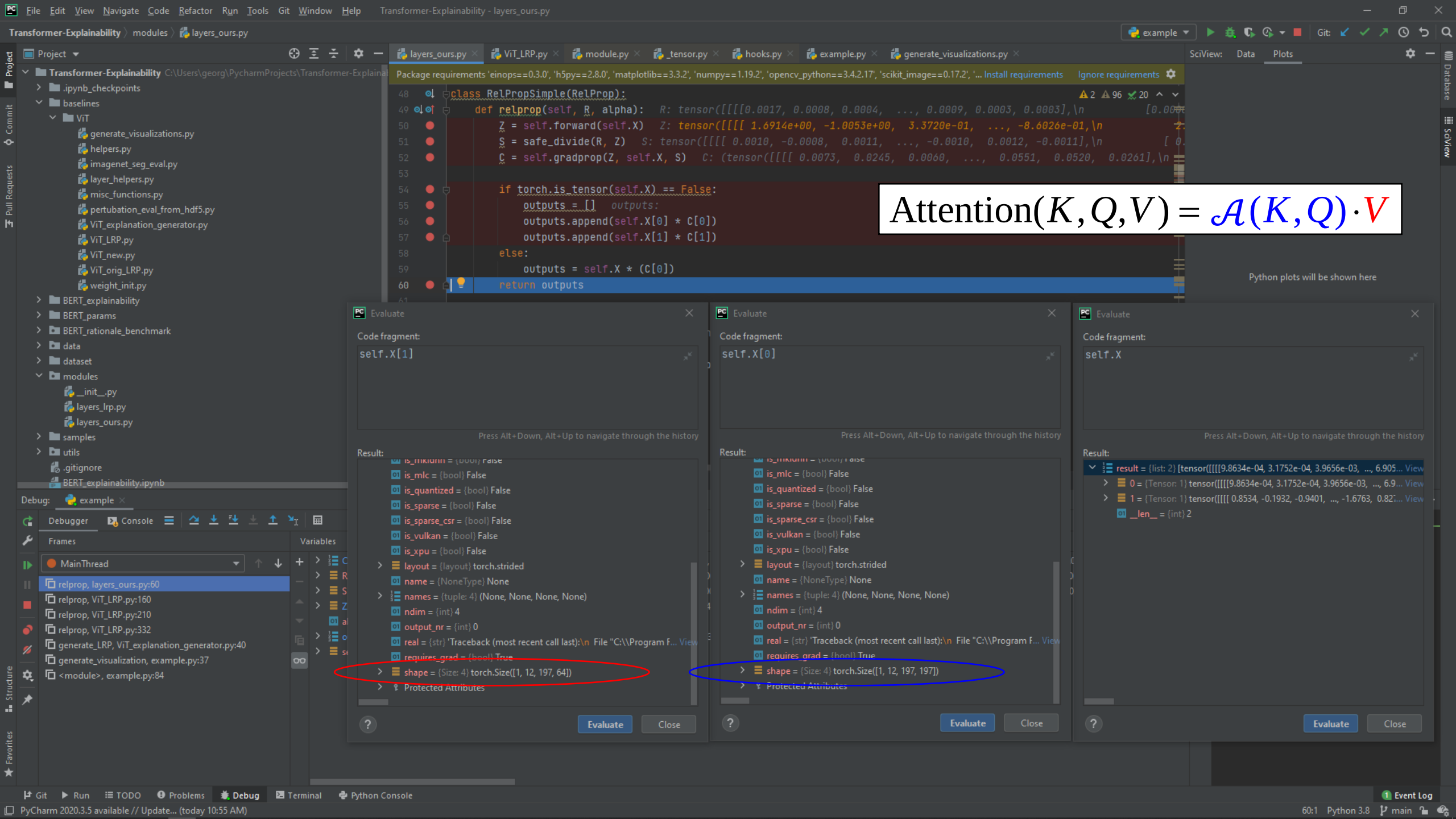

File Edit View Navigate Code Refactor Run Tools Git Window Help
Transformer-Explainability - ViT_LRP.py
Transformer-Explainability baselines ViT ViT_LRP.py
example
Git:
Project
Transformer-Explainability C:\Users\georg\PycharmProjects\Transformer-Explainabl
.ipynb_checkpoints
baselines
ViT
generate_visualizations.py
helpers.py
layers_ours.py
ViT_LRP.py
einops.py
example.py
generate_visualizations.py
Package requirements 'einops==0.3.0', 'h5py==2.8.0', 'matplotlib==3.3.2', 'numpy==1.19.2', 'opencv_python==3.4.2.17', 'scikit_image==0.17.2', 'scipy==1.5.2', 'sklearn', 'torch==1.7.0', 'torchvision==0.8.1', 'tqd... Install requirements Ignore requirements
111 def get_attn_cam(self):
112 return self.attn_cam
113
114 def get_v(self):
115 return self.v
Evaluate
Code fragment:
k
Press Alt+Down, Alt+Up to navigate through the history
Result:
is_mlc = {bool} False
is_quantized = {bool} False
is_sparse = {bool} False
is_sparse_csr = {bool} False
is_vulkan = {bool} False
is_xpu = {bool} False
layout = {layout} torch.strided
name = {NoneType} None
names = {tuple: 4} (None, None, None, None)
ndim = {int} 4
output_nr = {int} 1
real = {str} 'Traceback (most recent call last):\n File "C:\\Program F... View
requires_grad = {bool} True
shape = {Size: 4} torch.Size([1, 12, 197, 64])
Protected Attributes
Evaluate
Close
Evaluate
Code fragment:
q
Press Alt+Down, Alt+Up to navigate through the history
Result:
is_mlc = {bool} False
is_quantized = {bool} False
is_sparse = {bool} False
is_sparse_csr = {bool} False
is_vulkan = {bool} False
is_xpu = {bool} False
layout = {layout} torch.strided
name = {NoneType} None
names = {tuple: 4} (None, None, None, None)
ndim = {int} 4
output_nr = {int} 0
real = {str} 'Traceback (most recent call last):\n File "C:\\Program F... View
requires_grad = {bool} True
shape = {Size: 4} torch.Size([1, 12, 197, 64])
Protected Attributes
Evaluate
Close
Evaluate
Code fragment:
qkv
Press Alt+Down, Alt+Up to navigate through the history
Result:
is_mlc = {bool} False
is_quantized = {bool} False
is_sparse = {bool} False
is_sparse_csr = {bool} False
is_vulkan = {bool} False
is_xpu = {bool} False
layout = {layout} torch.strided
name = {NoneType} None
names = {tuple: 3} (None, None, None)
ndim = {int} 3
output_nr = {int} 0
real = {str} 'Traceback (most recent call last):\n File "C:\\Program F... View
requires_grad = {bool} True
shape = {Size: 3} torch.Size([1, 197, 2304])
Protected Attributes
Evaluate
Close
Evaluate
Code fragment:
x
Press Alt+Down, Alt+Up to navigate through the history
Result:
is_mlc = {bool} False
is_quantized = {bool} False
is_sparse = {bool} False
is_sparse_csr = {bool} False
is_vulkan = {bool} False
is_xpu = {bool} False
layout = {layout} torch.strided
name = {NoneType} None
names = {tuple: 3} (None, None, None)
ndim = {int} 3
output_nr = {int} 0
real = {str} 'Traceback (most recent call last):\n File "C:\\Program F... View
requires_grad = {bool} True
shape = {Size: 3} torch.Size([1, 197, 768])
Protected Attributes
Evaluate
Close
forward, ViT_LRP.py:198
_call_impl, module.py:1051
forward, ViT_LRP.py:316
_call_impl, module.py:1051
<module>, example.py:80
is_xpu = {bool} False
layout = {layout} torch.strided
name = {NoneType} None
names = {tuple: 4} (None, None, None, None)
ndim = {int} 4
output_nr = {int} 2
real = {str} 'Traceback (most recent call last):\n File "C:\\Program Files\\JetBrains\\PyC... View
requires_grad = {bool} True
shape = {Size: 4} torch.Size([1, 12, 197, 64])
Protected Attributes
x = {Tensor: 1} tensor([[[0.0217, 0.0427, -0.0051, ..., 0.0212, 0.0029, -0.0604],\n [0.007... View
Git TODO Problems Debug Terminal Python Console
Event Log
PyCharm 2020.3.5 available // Update... (47 minutes ago)
112:29 Python 3.8 main


File Edit View Navigate Code Refactor Run Tools Git Window Help Transformer-Explainability - ViT_LRP.py
Transformer-Explainability baselines ViT ViT_LRP.py
layers_ours.py ViT_LRP.py einops.py example.py generate_visualizations.py
Project
Transformer-Explainability C:\Users\georg\PycharmProjects\Transformer-Explainal
.ipynb_checkpoints
baselines
ViT
generate_visualizations.py
helpers.py
imagenet_seg_eval.py
layer_helpers.py
misc_functions.py
pertubation_eval_from_hdf5.py
ViT_explanation_generator.py
ViT_LRP.py
ViT_new.py
qkv = self.qkv(x)
q, k, v = rearrange(qkv, 'b n (qkv h d) -> qkv b h n d', qkv=3, h=h)
self.save_v(v)
dots = self.matmul1([q, k]) * self.scale
attn = self.softmax(dots)
attn = self.attn_drop(attn)
self.save_attn(attn)
attn.register_hook(self.save_attn_gradients)
Evaluate
Code fragment:
torch.matmul(q.cpu(), torch.transpose(k.cpu(), 2, 3)) * self.scale
Result:
result = {Tensor: 1} tensor([[[[5.1247e+00, -7.3943e-01, -6.5886e-01,
device = {device} cpu
dtype = {dtype} torch.float32
Code fragment:
q
shape = {Size: 4} torch.Size([1, 12, 197, 64])
requires_grad = {bool} True
Code fragment:
dots
result = {Tensor: 1} tensor([[[[5.1247e+00, -7.3943e-01, -6.5886e-01,
device = {device} cuda:0
dtype = {dtype} torch.float32
is_cuda = {bool} True
Code fragment:
attn
shape = {Size: 4} torch.Size([1, 12, 197, 197])
requires_grad = {bool} True
Evaluate Close
Git TODO Problems Debug Terminal Python Console
PyCharm 2020.3.5 available // Update... (today 10:55 AM)
Event Log
1:67 Python 3.8 main


File Edit View Navigate Code Refactor Run Tools Git Window Help Transformer-Explainability - ViT_LRP.py
Transformer-Explainability baselines ViT ViT_LRP.py
layers_ours.py
ViT_LRP.py
einops.py
example.py
generate_visualizations.py
Package requirements 'einops==0.3.0', 'h5py==2.8.0', 'matplotlib==3.3.2', 'numpy==1.19.2', 'opencv_python==3.4.2.17', 'scikit_image==0.17.2', 'scipy==1.5.2', 'sklearn', 'torch==1.7.0', 'torchvision==0.8.1', 'tqd... Install requirements Ignore requirements
Project
Transformer-Explainability C:\Users\georg\PycharmProjects\Transformer-Explainal
.ipynb_checkpoints
baselines
ViT
generate_visualizations.py
helpers.py
imagenet_seg_eval.py
layer_helpers.py
misc_functions.py
pertubation_eval_from_hdf5.py
ViT_explanation_generator.py
ViT_LRP.py
ViT_new.py
qkv = self.qkv(x) qkv: tensor([[[-0.5566, 0.1521, -0.8083, ..., -0.0709, 0.0237, -0.0271],\n [-2.0715, 0.4044, -1.7595,
q, k, v = rearrange(qkv, 'b n (qkv h d) -> qkv b h n d', qkv=3, h=h)
self.save_v(v)
dots = self.matmul1([q, k]) * self.scale dots: tensor([[[[5.1247e+00, -7.3943e-01, -6.5886e-01, ..., -1.0609e+00,\n -7.6978e-01, -3.1289e-01],\
attn = self.softmax(dots) attn: tensor([[[[7.5586e-01, 2.1463e-03, 2.3263e-03, ..., 1.5562e-03,\n 2.0821e-03, 3.2880e-03],\n [9.3099e-0
attn = self.attn_drop(attn)
self.save_attn(attn)
attn.register_hook(self.save_attn_gradients)
Evaluate
Code fragment:
torch.matmul(q.cpu(), torch.transpose(k.cpu(), 2, 3)) * self.scale
Press Alt+Down, Alt+Up to navigate through the history
Result:
result = {Tensor: 1} tensor([[[[5.1247e+00, -7.3943e-01, -6.5886e-01, ...
device = {device} cpu
dtype = {dtype} torch.float32
grad = {NoneType} None
grad_fn = {MulBackward0} <MulBackward0 object at 0x000001F2D357579
is_cuda = {bool} False
Evaluate
Close
Evaluate
Code fragment:
self.softmax(torch.matmul(q.cpu(), torch.transpose(k.cpu(), 2, 3)) * self.scale)
Result:
result = {Tensor: 1} tensor([[[[7.5586e-01, 2.1463e-03, 2.3263e-03, ..., 1.
device = {device} cpu
dtype = {dtype} torch.float32
grad_fn = {SoftmaxBackward} <SoftmaxBackward object at 0x000001F2D3
is_cuda = {bool} False
Evaluate
Code fragment:
dots
Result:
result = {Tensor: 1} tensor([[[[5.1247e+00, -7.3943e-01, -6.5886e-01, ...
device = {device} cuda:0
dtype = {dtype} torch.float32
grad_fn = {MulBackward0} <MulBackward0 object at 0x000001F2D36508B
is_cuda = {bool} True
Evaluate
Code fragment:
attn
Result:
result = {Tensor: 1} tensor([[[[7.5586e-01, 2.1463e-03, 2.3263e-03, ..., 1.
device = {device} cuda:0
dtype = {dtype} torch.float32
grad_fn = {SoftmaxBackward} <SoftmaxBackward object at 0x0000...
is_cuda = {bool} True
names = {tuple: 4} (None, None, None, None)
ndim = {int} 4
output_nr = {int} 2
requires_grad = {bool} True
Git TODO Problems Debug Terminal Python Console Event Log
Typo: In word 'bhij'
1:81 Python 3.8 main


File Edit View Navigate Code Refactor Run Tools Git Window Help
Transformer-Explainability - ViT_LRP.py
Transformer-Explainability baselines ViT ViT_LRP.py
example
Git:
Project
Transformer-Explainability C:\Users\georg\PycharmProjects\Transformer-Explainabi
.ipynb_checkpoints
baselines
ViT
generate_visualizations.py
helpers.py
imagenet_seg_eval.py
layer_helpers.py
misc_functions.py
pertubation_eval_from_hdf5.py
ViT_explanation_generator.py
ViT_LRP.py
ViT_new.py
layers_ours.py
ViT_LRP.py
_tensor.py
hooks.py
example.py
generate_visualizations.py
Package requirements 'einops==0.3.0', 'h5py==2.8.0', 'matplotlib==3.3.2', 'numpy==1.19.2', 'opencv_python==3.4.2.17', 'scikit_image==0.17.2', 'scipy==1.5.2', 'sklearn', 'torch==1.7.0', 'torchvision==0.8.1', 'tqd... Install requirements Ignore requirements
107
108 def save_attn_cam(self, cam):
109 self.attn_cam = cam
110
111 def get_attn_cam(self):
112 return self.attn_cam
113
114 def get_v(self):
115 return self.v
116
117 def save_v(self, v):
Evaluate
Code fragment:
torch.matmul(attn.cpu(), v.cpu())
Press Alt+Down, Alt+Up to navigate through the history
Result:
result = {Tensor: 1} tensor([[[[-2.2390e-02, -9.3515e-02, 6.0591e-02, ...,... View
T = {Tensor: 64} tensor([[[[-2.2390e-02],\n [-3.7934e-02],\n ... View
data = {Tensor: 1} tensor([[[[-2.2390e-02, -9.3515e-02, 6.0591e-02, ... View
device = {device} cpu
dtype = {dtype} torch.float32
grad = {NoneType} None
grad_fn = {UnsafeViewBackward} <UnsafeViewBackward object at 0x0000
imag = {str} 'Traceback (most recent call last):\n File "C:\\Program... View
is_cuda = {bool} False
is_leaf = {bool} False
is_meta = {bool} False
is_mkldnn = {bool} False
is_mlc = {bool} False
is_quantized = {bool} False
is_sparse = {bool} False
is_sparse_csr = {bool} False
Evaluate
Close
Evaluate
Code fragment:
torch.matmul(attn.cpu(), v.cpu())
Press Alt+Down, Alt+Up to navigate through the history
Result:
is_mlc = {bool} False
is_quantized = {bool} False
is_sparse = {bool} False
is_sparse_csr = {bool} False
is_vulkan = {bool} False
is_xpu = {bool} False
layout = {layout} torch.strided
name = {NoneType} None
names = {tuple: 4} (None, None, None, None)
ndim = {int} 4
output_nr = {int} 0
real = {str} 'Traceback (most recent call last):\n File "C:\\Program F... View
requires_grad = {bool} True
shape = {Size: 4} torch.Size([1, 12, 197, 64])
Protected Attributes
Evaluate
Close
Evaluate
Code fragment:
out
Press Alt+Down, Alt+Up to navigate through the history
Result:
result = {Tensor: 1} tensor([[[[-2.2390e-02, -9.3515e-02, 6.0591e-02, ...,... View
T = {Tensor: 64} tensor([[[[-2.2390e-02],\n [-3.7934e-02],\n ... View
data = {Tensor: 1} tensor([[[[-2.2390e-02, -9.3515e-02, 6.0591e-02, ... View
device = {device} cuda:0
dtype = {dtype} torch.float32
grad = {NoneType} None
grad_fn = {ViewBackward} <ViewBackward object at 0x00000223834EAE80
imag = {str} 'Traceback (most recent call last):\n File "C:\\Program... View
is_cuda = {bool} True
is_leaf = {bool} False
is_meta = {bool} False
is_mkldnn = {bool} False
is_mlc = {bool} False
is_quantized = {bool} False
is_sparse = {bool} False
is_sparse_csr = {bool} False
Evaluate
Close
Evaluate
Code fragment:
out
Press Alt+Down, Alt+Up to navigate through the history
Result:
is_mlc = {bool} False
is_quantized = {bool} False
is_sparse = {bool} False
is_sparse_csr = {bool} False
is_vulkan = {bool} False
is_xpu = {bool} False
layout = {layout} torch.strided
name = {NoneType} None
names = {tuple: 4} (None, None, None, None)
ndim = {int} 4
output_nr = {int} 0
real = {str} 'Traceback (most recent call last):\n File "C:\\Program F... View
requires_grad = {bool} True
shape = {Size: 4} torch.Size([1, 12, 197, 64])
Protected Attributes
Evaluate
Close
self = {Attention} Attention(\n (matmul1): einsum()\n (matmul2): einsum()\n (qkv): Line... View
v = {Tensor: 1} tensor([[[[-2.6453e-02, -1.1280e-01, 1.3936e-01, ..., -1.2810e-02,\n 6.... View
x = {Tensor: 1} tensor([[[[0.0217, 0.0427, -0.0051, ..., 0.0212, 0.0029, -0.0604],\n [0.00... View
Git Run TODO Problems Debug Terminal Python Console
Event Log
PyCharm 2020.3.5 available // Update... (today 10:55 AM)
1:34 Python 3.8 main


Concatenate all heads' outputs


File Edit View Navigate Code Refactor Run Tools Git Window Help
Transformer-Explainability - layers_ours.py
Transformer-Explainability 〉 modules 〉 layers_ours.py
example
Project
Transformer-Explainability C:\Users\georg\PycharmProjects\Transformer-Explainabi
.ipynb_checkpoints
baselines
ViT
generate_visualizations.py
helpers.py
imagenet_seg_eval.py
layer_helpers.py
misc_functions.py
pertubation_eval_from_hdf5.py
ViT_explanation_generator.py
ViT_LRP.py
ViT_new.py
ViT_orig_LRP.py
weight_init.py
BERT_explainability
BERT_params
BERT_rationale_benchmark
data
dataset
modules
__init__.py
layers_lrp.py
layers_ours.py
samples
utils
.gitignore
BERT_explainability.ipynb
layers_ours.py
ViT_LRP.py
module.py
_tensor.py
hooks.py
example.py
generate_visualizations.py
Package requirements 'einops==0.3.0', 'h5py==2.8.0', 'matplotlib==3.3.2', 'numpy==1.19.2', 'opencv_python==3.4.2.17', 'scikit_image==0.17.2', '... Install requirements Ignore requirements
class RelPropSimple(RelProp):
def relprop(self, R, alpha):
Z = self.forward(self.X)
S = safe_divide(R, Z)
C = self.gradprop(Z, self.X, S)
if torch.is_tensor(self.X) == False:
outputs = []
outputs.append(self.X[0] * C[0])
outputs.append(self.X[1] * C[1])
else:
outputs = self.X * (C[0])
return outputs
SciView: Data Plots
Python plots will be shown here
Attention(K, Q, V) = A(K, Q) · V
Evaluate
Code fragment:
self.X[1]
Press Alt+Down, Alt+Up to navigate through the history
Result:
is_mlc = {bool} False
is_quantized = {bool} False
is_sparse = {bool} False
is_sparse_csr = {bool} False
is_vulkan = {bool} False
is_xpu = {bool} False
layout = {layout} torch.strided
name = {NoneType} None
names = {tuple: 4} (None, None, None, None)
ndim = {int} 4
output_nr = {int} 0
real = {str} 'Traceback (most recent call last):\n File "C:\\Program F... View
requires_grad = {bool} True
shape = {Size: 4} torch.Size([1, 12, 197, 64])
Protected Attributes
Evaluate
Close
Evaluate
Code fragment:
self.X[0]
Press Alt+Down, Alt+Up to navigate through the history
Result:
is_mlc = {bool} False
is_quantized = {bool} False
is_sparse = {bool} False
is_sparse_csr = {bool} False
is_vulkan = {bool} False
is_xpu = {bool} False
layout = {layout} torch.strided
name = {NoneType} None
names = {tuple: 4} (None, None, None, None)
ndim = {int} 4
output_nr = {int} 0
real = {str} 'Traceback (most recent call last):\n File "C:\\Program F... View
requires_grad = {bool} True
shape = {Size: 4} torch.Size([1, 12, 197, 197])
Protected Attributes
Evaluate
Close
Evaluate
Code fragment:
self.X
Press Alt+Down, Alt+Up to navigate through the history
Result:
result = {list: 2} [tensor([[[[9.8634e-04, 3.1752e-04, 3.9656e-03, ..., 6.905... View
0 = {Tensor: 1} tensor([[[[9.8634e-04, 3.1752e-04, 3.9656e-03, ..., 6.9... View
1 = {Tensor: 1} tensor([[[[0.8534, -0.1932, -0.9401, ..., -1.6763, 0.82... View
__len__ = {int} 2
Evaluate
Close
Debug: example
Debugger Console
Frames
Variables
MainThread
relprop, layers_ours.py:60
relprop, ViT_LRP.py:160
relprop, ViT_LRP.py:210
relprop, ViT_LRP.py:332
generate_LRP, ViT_explanation_generator.py:40
generate_visualization, example.py:37
<module>, example.py:84
Git Run TODO Problems Debug Terminal Python Console
Event Log
PyCharm 2020.3.5 available // Update... (today 10:55 AM)
60:1 Python 3.8 main