

# Google Universal Image Embedding Challenge

EEE511(Fall 22,Team#9): Sumant Kulkarni, Dhanraj Bhosale, Pratyush Pandey, Shayal Shamsu

- Hosted by Kaggle in collaboration with Google Research and Google Lens.
- Developed models are expected to retrieve relevant database images for a given query image
- Challenge is to build a single universal image embedding model capable of representing objects from multiple domains at the instance level.
- Image dataset comprises a variety of object types - Apparel, Artwork, Landmarks, Furniture, & Packaged Goods, among others.



# Expected Results and Datasets

## ➤ Expected Results

- We will submit the developed model in the Kaggle competition. The model will be evaluated against a held out test datasets and a score will be assigned.
- We will build a simulation jig to demonstrate the model performance. The model will be provided a set of input images and it will search through a dataset of images and identify images which match the input images. We will demonstrate with multiple images from different object types.
- It would be exciting for all class to see implementation of one of the top DL competition.

## ➤ Datasets

- Imagenet - <https://www.image-net.org/index.php> (Available for free to researchers for non-commercial use)
- Products -10K - <https://products-10k.github.io/> (Available for free for non-commercial research and educational purposes)
- Google Landmark Recognition 2021 - <https://www.kaggle.com/competitions/landmark-recognition-2021/data> (Dataset is part of Kaggle competition in 2021)
- Some datasets are provided by Google as part of the competition
- 90% data is used for training while 10% is used for validation.

# Key Literature and Implementation Code

## ➤ Key Literature

- Google AI Blog - Introducing the Google Universal Image Embedding Challenge, August 4, 2022, Posted by Bingyi Cao and Mário Lipovský, Software Engineer, Google Lens, <https://ai.googleblog.com/2022/08/introducing-google-universal-image.html>
- Baseline model implementation for the Kaggle universal image embedding challenge - [https://github.com/google-research/google-research/tree/master/universal\\_embedding\\_challenge](https://github.com/google-research/google-research/tree/master/universal_embedding_challenge)
- Training data-efficient image transformers & distillation through attention - <https://arxiv.org/pdf/2012.12877.pdf>
- Transformers for image recognition at scale - <https://arxiv.org/pdf/2010.11929.pdf>

## ➤ Implementation Code

- We are using the following Code notebook as a reference implementation - <https://www.kaggle.com/code/akihirok/9th-place-guie-fintune-tf-clip-with-training>
- The notebook has achieved a score amongst the top 10 performers in the competition and is most used and commented by the users.
- For training: backbone(CLIP) + Dropout + Dense(units=256) + Arcface + Softmax (classes=17691)
- For inference: backbone(CLIP) + Dropout + Dense(units=64) + L2Norm

# Computation Platform and Progress

- We are using Google Colaboratory, Kaggle Notebook, Tensorflow and TPUs for the training and the development of the model. All the needed software environment is available in Google Colaboratory.
- We have replicated the entire notebook and have **successfully developed** the model that can be submitted for the competition. It includes running several epochs of training.
- **Link of Video Clip of Model Training –**  
<https://drive.google.com/file/d/1pYY6dtD1uJMVzEj-woFSRSw08WmqBDzE/view?usp=sharing>
- We are going to develop the simulation test rig on our individual systems to demonstrate the model.

```
emb_model.summary()
```

Model: "sequential\_1"

Layer (type)	Output Shape	Param #
lambda_1 (Lambda)	(None, None, None, 3)	0
resize (Lambda)	(None, 224, 224, 3)	0
permute_1 (Permute)	(None, 3, 224, 224)	0
model_2 (Functional)	(None, 1280)	630766080
dropout_65 (Dropout)	(None, 1280)	0
dense_1 (Dense)	(None, 64)	81984
embedding_norm (Lambda)	(None, 64)	0
Total params: 630,848,064		
Trainable params: 630,848,064		
Non-trainable params: 0		

Device

```
import tensorflow as tf
try:
    tpu = tf.distribute.cluster_resolver.TPUCacheResolver()
    print('Running on TPU ', tpu.master())
except ValueError:
    tpu = None

if tpu:
    tf.config.experimental_connect_to_cluster(tpu)
    tf.tpu.experimental.initialize_tpu_system(tpu)
    strategy = tf.distribute.TPUStrategy(tpu)
else:
    # Default distribution strategy in TensorFlow. Works on CPU and single GPU.
    strategy = tf.distribute.get_strategy()

AUTO = tf.data.experimental.AUTOTUNE
print('REPLICAS: ', strategy.num_replicas_in_sync)
```

Running on TPU /CPU:0.2.0470

```
2022-10-13 18:18:51.594229: I tensorflow/core/distributed_runtime/rpc/grpc_channel.cc:381] Initialize GrpcChannelCache for job worker -> (0 -> 10.0.0.2:8470)
2022-10-13 18:18:51.594447: I tensorflow/core/distributed_runtime/rpc/grpc_channel.cc:381] Initialize GrpcChannelCache for job localhost -> (0 -> localhost:3
)
2022-10-13 18:18:51.598380: I tensorflow/core/distributed_runtime/rpc/grpc_channel.cc:381] Initialize GrpcChannelCache for job worker -> (0 -> 10.0.0.2:8470)
2022-10-13 18:18:51.598384: I tensorflow/core/distributed_runtime/rpc/grpc_channel.cc:381] Initialize GrpcChannelCache for job localhost -> (0 -> localhost:3
)
REPLICAS: 0
```

```
print('# of ttfrecords for training : ', len(train_set_path))
print('# of ttfrecords for validation : ', len(valid_set_path))
```

```
if config.DEBUG:
    train_set_path = random.sample(train_set_path, 4)
    print('debug: reduce training data. num=', len(train_set_path))

    valid_set_path = train_set_path + valid_set_path[:1]
    print('debug: reduce validation data. num=', len(valid_set_path))
```

```
"guie-g/20221013-ttfrecords-label-10692-17690" : "gs://tds-4f2615c7b959d3b3e6e6c774b68124e8170804e6f69f55f4ec0a",
guie-g/20221013-ttfrecords-label-10692-17690, number of ttfrecords = 32
```

```
2022-10-13 18:18:58.040740: I tensorflow/core/platform/cloud/google_auth_provider.cc:180] Attempting an empty bearer token since no token was retrieved from file
s, and GCE metadata check was skipped.
```

```
"guie-unguestitl-minil-ttfrecords-label-0-000" : "gs://tds-4f2615c7b959d3b3e6e6c774b68124e8170804e6f69f55f4ec0a",
guie-unguestitl-minil-ttfrecords-label-0-000, number of ttfrecords = 50
```

```
2022-10-13 18:18:58.337779: I tensorflow/core/platform/cloud/google_auth_provider.cc:180] Attempting an empty bearer token since no token was retrieved from file
s, and GCE metadata check was skipped.
```

```
"guie-products30k-ttfrecords-label-1000-10690" : "gs://tds-4f2615c7b959d3b3e6e6c774b68124e8170804e6f69f55f4ec0a",
guie-products30k-ttfrecords-label-1000-10690, number of ttfrecords = 20
```

```
# of ttfrecords for training : 91
```

```
# of ttfrecords for validation : 11
```

```
2022-10-13 18:18:58.684805: I tensorflow/core/platform/cloud/google_auth_provider.cc:180] Attempting an empty bearer token since no token was retrieved from file
s, and GCE metadata check was skipped.
```

# Remaining Activities

- Revise the developed model with additional training datasets and epochs – All team members – Ongoing, to be completed by Nov-11
- Submit the model to Kaggle competition and obtain the evaluation score – Dhanraj and Pratyush – First submission by Oct-21, several submissions based on improvements in the model
- Develop the test rig to demonstrate the model – Sumant and Shayal - by Oct-28
- Document Progress Check 2 – Sumant and Dhanraj – Nov-11
- Develop the Project presentation – All team members – Nov-11 to Nov-15
- Document the First draft Project report – All team members – Nov-1 to Nov-15
- Group Review of the Project report – All team members – Nov-16 to Nov-23
- Document and submit the Final Project report – All team members – Nov-23 to Dec-9