**Project background**

CBRE has thousands of pictures of flexible spaces around the globe and wants to better match the demand and supply based on the understanding of clients’ preferences on style.

For the team, our tasks are to:

* Test the idea of identifying and matching styles
* Try different models to get latent style representations
* Design test prototypes for the product

We then divided into two teams:

1. Product team: follow the Sprint Method, use quick tests to cut down the risks, design prototypes (Yajing, Brian)
2. Technical team: collect and clean sample data, test models (Xuelan, Maria, Yiyuan)

**Finished work**

*Product team:*

1. Figure out Spring questions & identify all the risks → pick up most important questions:
   1. Do people care about style when making decisions?
   2. If so, how do they consider style factors when selecting offices?
2. Design tests to test main questions & mitigate risks
   1. Test 1: design an interactive prototype to test whether the users care about style
   2. Test 2: design a survey with open-ended questions to explore users and figure out what they consider when selecting offices
3. Help the technical team to test major assumptions:

* Hypothesis: pics from the same office have similar styles and people could recognize the similarities.
* Test 3: design surveys to test whether people could recognize pictures from the same office (if they could → same office have similar styles)

*Technical:*

1. Data collection: collected pictures from 6 suppliers, which were Wework, Industrious, Neuehouse, Office Spaces, Premier Workspaces and VentureX, and created a sample dataset containing 7479 pictures in total
2. Image type classification: built a CNN model to label whether an image is “Indoor”, “Outdoor” or “Floor Plan”
3. Style encoding:

We tested 4 possible solutions:

1. Use transfer learning. Inject a pre-trained model to generate flattened layers and save them as the style-related features
2. Transform to a self-supervised problem. Use autoencoder to learn a compressed representation of raw images
3. Use transfer learning + Autoencoder. The basic ideas are that assuming “image=content + style” and “the mature pretrained model like ResNet can capture content features as they are trained to classify the object in an image”, so we can learn the style-related features by combining an untrained model with the pretrained model to regenerate the original image
4. Transform to a supervised problem by creating labels or using proxy labels like producer or office. Enlightened by the frontier contrastive learning model “simclr”, we do data augmentation on 4 selected images and use the “augmented” images as input and which original image it belongs to as output. The basic idea is that the “augmented” images from the same original image should have the most similar style

**Helpful results**

* People do care about style when making decisions
* People have different style preferences over different functional areas
* Different offices contain different styles that ordinary people can identify
* Methods do not fit our problem: directly using pre-trained models like ResNet or using autoencoder method
* The potentially best suitable model for the style identification problem: *SimCLR*
* Potential evaluation methods for this unsupervised problem:
  + Check if the most similar and most dissimilar images in a test set found based on the representations make sense
  + Generate a simple test set by labeling the most similar image for each image and evaluate the representations’ performance on predicting the most similar image

**Index for files**

* Data
* Biweekly meeting decks
* Notebooks
* Docs
* Product team tests & prototypes