Assignment 6.3: ResNet50

In [58]:

```
import numpy as np
import matplotlib.pyplot as plt
import os, os.path
import sys

from PIL import Image

from tensorflow.keras.applications.resnet50 import ResNet50
from tensorflow.keras.preprocessing import image
from tensorflow.keras.applications.resnet50 import preprocess_input, decode_
```

Importing Images

In [37]:

```
imgs = []
names = []
img_path = 'images'

for f in os.listdir(img_path):
    imgs.append(Image.open(os.path.join(img_path,f)))
    names.append(f)
```

Converting Images

In [26]:

```
conv_imgs = []

for i in imgs:
    img = i.resize((224,224))
    x = image.img_to_array(img)
    x = np.expand_dims(x, axis=0)
    x = preprocess_input(x)
    conv_imgs.append(x)
```

Importing the Model

In [19]:

```
model = ResNet50(weights='imagenet')
```

```
Downloading data from https://storage.googleapis.com/tensorflow/keras-appli
cations/resnet/resnet50_weights_tf_dim_ordering_tf_kernels.h5
102973440/102967424 [==============================] - 32s 0us/step
102981632/102967424 [==============================] - 32s 0us/step
```

Predictions

In [27]:

```
preds = []

for i in conv_imgs:
    pred = model.predict(i)
    preds.append(pred)
```

In [45]:

```
for p in range(0,10):
    x = decode_predictions(preds[p], top=3)[0]
    print('Album:', names[p].split('.')[0])
    for item in x:
        if len(item[1]) > 8:
            print('Item:', item[1], '\tMatch: ', item[2])
        else:
            print('Item:', item[1], '\t\tMatch: ', item[2])
    print('\n')
```

```
Album: redTV
Item: trench_coat       Match:  0.387972
Item: cloak             Match:  0.22810045
Item: sweatshirt        Match:  0.043515887


Album: lover
Item: miniskirt         Match:  0.32951158
Item: jersey            Match:  0.17070642
Item: pajama            Match:  0.12040079


Album: midnights
Item: web_site          Match:  0.4937468
Item: lipstick          Match:  0.40501288
Item: iPod              Match:  0.016202198


Album: 1989
Item: book_jacket       Match:  0.5677164
Item: web_site          Match:  0.15202771
Item: envelope          Match:  0.055822678


Album: folklore
Item: park_bench        Match:  0.47377983
Item: maze              Match:  0.099759296
Item: megalith          Match:  0.055295918


Album: speaknowTV
Item: feather_boa       Match:  0.9827891
Item: gown              Match:  0.011839051
Item: wig               Match:  0.0017972493


Album: evermore
Item: fur_coat          Match:  0.18469208
Item: fountain          Match:  0.17546014
Item: caldron           Match:  0.104735166


Album: debut
Item: bikini            Match:  0.267524
Item: wig               Match:  0.1331575
Item: maillot           Match:  0.0903908


Album: fearlessTV
Item: hair_spray        Match:  0.66192985
Item: microphone        Match:  0.03473998
Item: harmonica         Match:  0.025322389


Album: reputation
Item: jersey            Match:  0.31122938
```

```
Item: web_site          Match:  0.08889243
Item: Band_Aid          Match:  0.0838832
```

Saving Predictions

In [ ]:

```python
with open('results/predictions/resnet50', 'w') as f:
    sys.stdout = f
    for p in range(0,10):
        x = decode_predictions(preds[p], top=3)[0]
        print('Album:', names[p].split('.')[0])
        for item in x:
            if len(item[1]) > 8:
                print('Item:', item[1], '\tMatch: ', item[2])
            else:
                print('Item:', item[1], '\t\tMatch: ', item[2])
        print('\n')
```