Recognition of features of faces on CelebA database with Variational Autoencoders

(Az arckép jellemzőinek a felismerése CelebA adatbázison variációs autoenkóderek segítségével)

Abstract

After a short discovery of the CelebA database, a well-known discriminative model was fit to learn the labels of the CelebA database: a pretrained Inception V3 [7] was used with transfer learning with a proper validation accuracy. A massive data augmentation was used to boost the performance.

Then, a generative model, a variational autoencoder was trained to learn the features of the faces. Here, a so-called latent traversal was performed to visualize the meaning of the dimensions. Fortunately, many of the 64 dimensions seemed to be meaningful and could learn some features of the faces (i.e.: color of hair, existence of bangs, color of the skin, etc), however the disentanglement requirement could not be achieved with these computational resources.

In Hungarian: A CelebA adatbázist néhány alapvető elemzés (pl.korreláció vizsgálat) elvégzése után tanító, teszt és validációs adatbázisra osztottuk ügyelve a kiegyesúlyozott felosztásra. Ezután egy jól ismert diszkriminatív modelt használtunk az adatbázis címkéinek megtanítására. Az InceptionV3 előtanított hálót használtuk erre a célra és a pontosság növelése érdekében adatdúsítást alkalmaztunk. Ennél jóval izgalmasabb kérdésnek ígérkezett a generatív model illesztése: variációs autoenkódert használtunk, amely képes volt megtanulni az arc egyes jellemzőit és ennek megfelelően új képeket generálni a látensek bejárásával. A látensek nagy része jól interpretálható jelentést hordoz (pl. hajszín, frufru, stb), ám a disentangled tulajdonságot jelen modellel és főleg számítási kapacitással nem teljesítettük teljes mértékben.

Introduction

The features of images in a labelled dataset can be learned by using discriminative models with convolutional neural networks, while in case of unlabelled image set several unsupervised techniques are proposed. Although with GANs ([11]) have great advantages in generating new images and find implicit latents, working with Variational Autoencoders was preferred as (i) inference and the regularisation of the latent space can be directly controlled (ii) posterior inference (which might be intractable) can be made efficient by fitting an approximate inference.

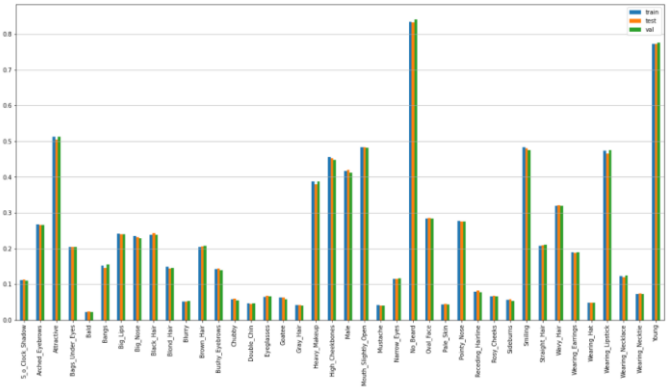
Here a conditional variational autoencoder was used (hierarchical VAE was not attempted finally but we are looking forward to have the time for it.)

Steps

Discovery of the dataset and determine the training / test / validation dataset

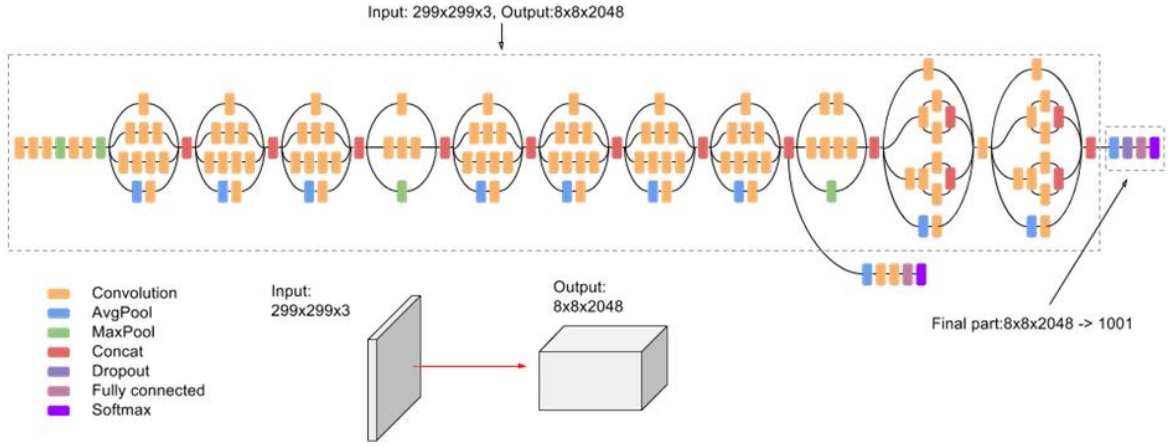
The balance of the dataset is proven to be essential as shown in the following documents [8-9]

I.e. generally, the distribution of the training data has a huge impact on the performance of CNN. The balanced distribution yielded a significantly better performance than imbalanced one. The heavier the imbalance is, the worse the total classification performance. This kind of fragility when using imbalanced data in the CNN training algorithm can be eliminated by the proper selection of distribution of dataset. Based on these points a splitting function and a check for distribution of data was performed.



We also investigated some solutions for dataset generation and preprocession. [A,B,C,D].To discover the relationship between the attributes we plotted the correlation matrix between the values.

Then a pretrained network, i.e. an Inception V3 was trained to learn the labels of the dataset.



Note, that those team members who would have been responsible for training hyperparameter tuning and visualization quitted the from the group.

Generative models

Short summary of variational autoencoders

To observe the intrinsic structure of a dataset auto-encoders can be very efficient by finding a right pair of encoders and decoders keeping the maximum of information when encoding and the minimum reconstruction error when decoding. However, the lack of interpretable structures in the latent space (a.k.a lack of regularity of the latent space) and overfitting (because of the reconstruction component in the loss function) can result in severe inadequacy. I.e. neither continuity (two close points in the latent space should not give two completely different contents once decoded) nor completeness (for a chosen distribution, a point sampled from the latent space should give “meaningful” content once decoded) can be guaranteed.

In variational autoencoder [1,2] we prevent the model to encode data far apart in the latent space and encourage as much as possible returned distributions to “overlap” with regularisation term, satisfying this way the expected continuity and completeness conditions.

Mathematical background of VAEs

Let us examine a latent variable model given by:

$$p(x,z) = p(z)p(x|z)$$

here we denoted the observed data by x and the latent variable with z. p(z) is called the prior distribution, and p(x|z) is called likelihood of our data x given the latent z.

In Bayesian statistics the computation of the posterior distribution is called as inference problem, where p(x) is the marginal probability. It is also called evidence of the data and can be calculated as the integral of the p(x,z).

If the integral does not have an analytical solution it can be intractable. In these cases we can find to try a proper approximation from a distribution family. The similarity is measured with Kullback - Leibler divergence between the approximation and the true posterior distribution.

$$\log p_\theta(x) = \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x)] \tag{3}$$
$$= \mathbb{E}_{q_\phi(z|x)}\left[\log\left(\frac{p_\theta(x,z)}{p_\theta(z|x)}\right)\right] \tag{4}$$
$$= \mathbb{E}_{q_\phi(z|x)}\left[\log\left(\frac{p_\theta(x,z)}{q_\phi(z|x)}\frac{q_\phi(z|x)}{p_\theta(z|x)}\right)\right] \tag{5}$$
$$= \mathbb{E}_{q_\phi(z|x)}\left[\log\left(\frac{p_\theta(x,z)}{q_\phi(z|x)}\right)\right] + \mathbb{E}_{q_\phi(z|x)}\left[\log\left(\frac{q_\phi(z|x)}{p_\theta(z|x)}\right)\right] \tag{6}$$

In case of optimization of objective, we use the so-called evidence lower bound (ELBO) which is typical in other variational methods. Here a sum of (6) is the ELBO and second one is the Kullback - Leibler divergence.

Note that Kullback - Leibler divergence is non-negative, which means that ELBO is a lower bound on the likelihood of the data. Thus, we we can maximize the ELBO in order to achieve two things at the same time: maximize the probablility of our data and minimize the KL divergence between the true and the approximated posteriori distribution.

To compute the approximate posterior q, we can design a neural network with parameters ϕ, called the encoder $q_\phi(z|x)$. In order to reconstruct the data, we can use another neural network with parameters θ, which is represented as $p_\theta(x|z)$, referred as the decoder. We optimize the so called variational parameters ϕ such that $q_\phi(z|x) \approx p_\theta(z|x)$.

Reparametrization trick is also used in our code (see class CVAE in forward function).

In the background we should consider the following mathematical problem:

In order to backpropagate through $z \sim q_\phi(z|x)$, if $z \sim N(\mu,\sigma 2)$, we have to write z in the form: $z \sim \mu + \sigma \cdot N(0,1)$. The reason is that as the model uses random sampling and we cannot backpropagate through a random note, we need some trick to overcome this problem. By separating the randomness, we are able to compute the gradienst using μ and σ.

The great novelty of Variational Autoencoders was that they combined probabilistic models, i.e. variational Bayesian approach with deep learning techniques so that to perform efficient inference and continuous latent variables even in case of intractable posterior distributions, and large datasets.

With a reparameterization of the variational lower bound, a differentiable unbiased estimator of the lower bound can be achieved: the SGVB (Stochastic Gradient Variational Bayes) estimator can be used for efficient approximate posterior inference.

Implementation

In PyTorch framework a conditional VEA was performed based on [E-F]. As own contribution (i) latent traversal, (ii) hyperparameter optimalisation (iii) callbacks (iv) dimension evaluation (v) comparison between original and reconstructed images were implemented. Different network size (parameter k) and different batch size (parameter b) were run with early stopping callback. With fixed random seeds the checkpoints were saved and upload to the Results folder. The best results had to be selected manually as all the latent traversal had to be

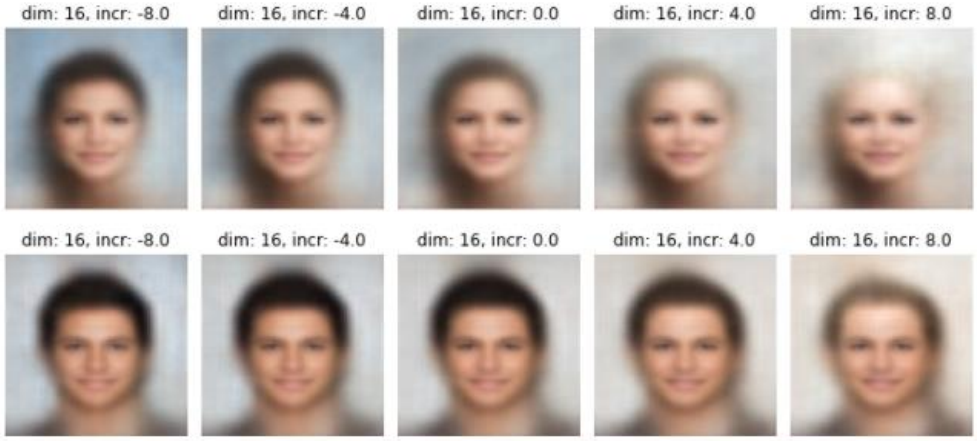
compared in all 64 dimension and meaningful latents were searched on different faces. File vae_best.pt was selected and upload to Google drive so as to be able to download it in the notebook (for your convenience).

Results

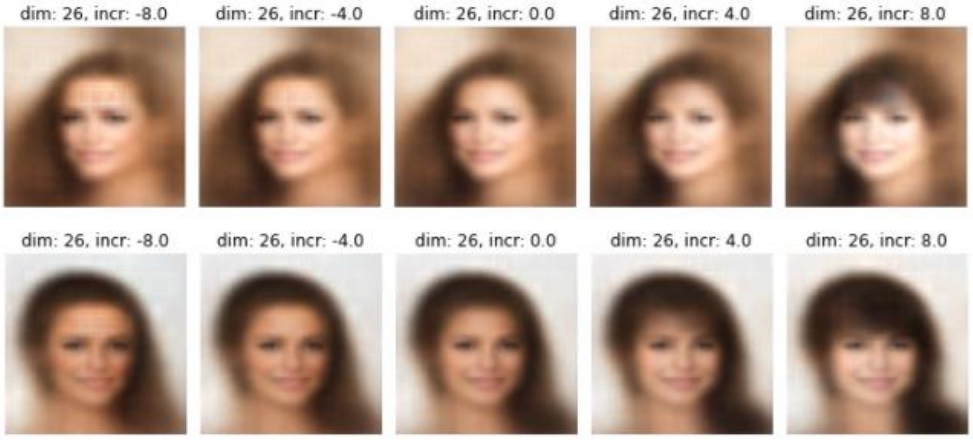
Many of the latents were meaningful. Note that according to the random seed and from different network size (parameter k) and batchsize (parameter b) different results were found.

Here you can see the best results:

Color of the hair at dimension 16th



Existence of bangs at dimension 26th

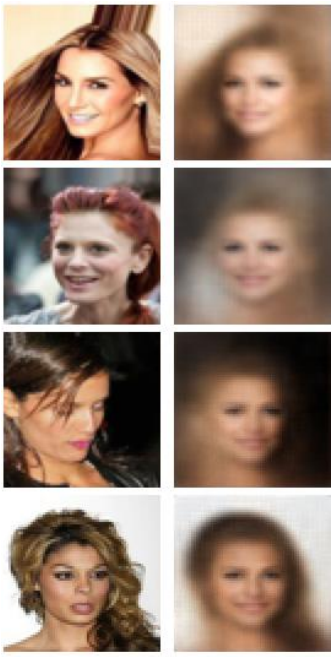


The rotation of face at dimension 2nd dimension

The thickness of hair at dimension 9th dimension

The width of the face at 12th dimension

Comparison of original and reconstructed images



Future plans

Hierarchical variational autencoders should be trained to achieve a more accureate solution (for the details see Future Planes documentation)

My future plans is to run a hierarchical variational autoencoder on CelebA dataset as it is described below.

The great novelty of Variational Autoencoders was that they combined probabilistic models, i.e. variational Bayesian approach with deep learning techniques so that to perform efficient inference and continuous latent variables even in case of intractable posterior distributions, and large datasets.

With a reparameterization of the variational lower bound, a differentiable unbiased estimator of the lower bound can be achieved: the SGVB (Stochastic Gradient Variational Bayes) estimator can be used for efficient approximate posterior inference.

However, the interpretability of automatedly discovered factorized representation of the independent data generative factors needed further modification on VAEs, i.e.: the introduction of an adjustable hyperparameter β that balances latent channel capacity and independence constraints
with reconstruction accuracy. β-VAE with $\beta > 1$ outperforms VAEs when the parameter was appropriately tuned [3.]. This modification limits

the capacity of latent variable, which, combined with the pressure to maximize the log likelihood of the training data, should encourage the model to learn the most efficient representation of the data as the Kubleck-Leibler divergence term of the β-VAE objective function encourages conditional independence in the posterior: higher values of β should encourage learning a disentangled representation.

Quantitatively comparing the different unsupervised deep generative models a crucial point is the degree of disentanglement of the latent variables. In case of a disentangled representation one single latent units are sensitive to changes in single generative factors, while being relatively invariant to changes in other factors [4.] and knowledge about one factor can generalize to novel configurations of other factors. [4.] also provides a protocol to quantitatively compare the degree of disentanglement learnt by different models.

Log likelihood of the data under the learnt model is a poor metric for evaluating disentangling in β-VAEs as latent channel capacity restriction (β > 1) can lead to poorer reconstructions due to the loss of high frequency details when passing through a constrained latent bottleneck. [3.] propose a quantitative metric that directly measures the degree of learnt disentanglement in the latent representation. It is important to note that a representation consisting of independent latents is not necessarily disentangled: i.e. PCA or ICA do not in general align with the data generative factors and hence may lack interpretability thus a simple cross-correlation calculation between the inferred latents would not suffice as a disentanglement metric.

Instead: inference is run on a number of images that are generated by fixing the value of one data generative factor while randomly sampling all others.

A low capacity linear classifier is used to identify the fixed factor and report the accuracy value as the final disentanglement metric score (as the independence and interpretability properties hold for the inferred representations, thus there will be less variance in the inferred latents that correspond to the fixed generative factor. Smaller variance in the latents corresponding to the target factor will make the job of this classifier easier, resulting in a higher score under the metric.

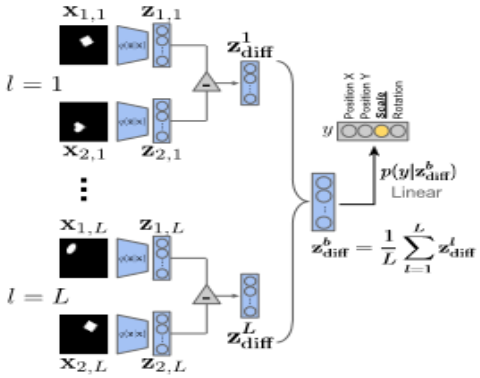


Figure 5: Schematic of the proposed disentanglement metric: over a batch of L samples, each pair of images has a fixed value for one target generative factor y (here $y = scale$) and differs on all others. A linear classifier is then trained to identify the target factor using the average pairwise difference $\mathbf{z}^b_{\text{diff}}$ in the latent space over L samples.

Tuning the β coefficient

β can be considered as a mixing coefficient for balancing the magnitudes of gradients from the reconstruction and the prior-matching components of the VAE lower bound formulation. β should be normalized by latent z size m and input x size n in order to compare its different values across different latent layer sizes and different datasets (βnorm=β*M/N). We found that larger latent z layer sizes m requires higher constraint pressures (higher β values). Furthermore, the relationship of β for a given m is characterised by an inverted U curve. When β is too low or too high the model learns an entangled latent representation due to either too much or too little capacity in the latent z bottleneck: β > 1 is necessary to achieve good disentanglement. However, if β is toohigh and the resulting capacity of the latent channel is lower than the number of data generative factors, then the learnt representation necessarily has to be entangled (as a low-rank projection of the true data generative factors will compress them in a non-factorial way to still capture the fulldata distribution well).

We proposed a principled way of choosing β for datasets with at least weak label information. If label information exists for at least a small subset of the independent data generative factors of variation, one can apply the disentanglement metric described in Sec. 3 to approximate the level of learnt disentanglement for various β choices during a hyperparameter sweep.

Hierarchical VAEs

A more sophisticated approach for image generation is the nouveau VAE (NVAE) [5], a deep hierarchical VAE using depth-wise separable convolutions and batch normalization. NVAE is equipped with a residual parameterization of normal distributions and its training is stabilized by spectral regularization. NVAE uses depthwise convolutions in its generative model with (i) a new residual

parameterization of the approximate posteriors. ii) stabilized training deep VAEs with spectral regularization, iii) practical solutions are used to reduce the memory burden of VAEs. iv) deep hierarchical VAEs can obtain state-of-the-art results on several image datasets and can produce high-quality samples even when trained with the original VAE objective.
The main building block of NVAE is depthwise convolutions that rapidly increase the receptive field of the network without dramatically increasing the number of parameters. (In depth-wise convolution, we use each filter channel only at one input channel.) NVAE was the first successful application of VAEs to images as large as 256×256 pixels.

References of literature

[1.] Diederik P Kingma and Max Welling.
Auto-encoding variational bayes.
In The International Conference on Learning Representations (ICLR), 2014.
https://arxiv.org/pdf/1312.6114.pdf

[2.] Diederik P Kingma and Max Welling.
An introduction to variational autoencoders
Foundations and Trends® in Machine Learning, 2019.
https://arxiv.org/pdf/1906.02691.pdf

[3.] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, Alexander Lerchner
ICLR 2017 conference submission, 2017.
Beta-vae: Learning basic visual concepts with a constrained variational framework
https://openreview.net/forum?id=Sy2fzU9gl

[4.] Y. Bengio, A. Courville, and P. Vincent.
Representation learning: A review and new perspectives. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2013

[5.] Arash Vahdat, Jan Kautz
NVAE: A Deep Hierarchical Variational Autoencoder
https://arxiv.org/pdf/2007.03898.pdf

[6.] Tero Karras, Timo Aila, Samuli Laine, Jaakko Lehtinen Progressive Growing of GANs for Improved Quality, Stability, and Variation
https://arxiv.org/pdf/1710.10196v3.pdf

[7.] SZEGEDY, Christian, et al. Going deeper with convolutions.

In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2015. p. 1-9.
Going Deeper With Convolutions (googleusercontent.com)

[8.] *Tang (May 2018).* Intelligent Mobile Projects with TensorFlow*. Packt Publishing. pp. Chapter 2.* ISBN 9781788834544*.*

[9] Karim and Zaccone (March 2018). Deep Learning with TensorFlow. Packt Publishing. pp. Chapter 4. ISBN 9781788831109.

[10] M*ilton-Barker, Adam.* "Inception V3 Deep Convolutional Architecture For Classifying Acute Myeloid/Lymphoblastic Leukemia". *intel.com. Intel*. Retrieved 2 February 2019*.*

[11]

Tero Karras, Timo Aila, Samuli Laine, Jaakko Lehtinen

Progressive Growing of GANs for Improved Quality, Stability, and Variation

ICLR 2018 Conference Blind Submission

Progressive Growing of GANs for Improved Quality, Stability, and Variation | OpenReview

Some essays, tutorials, codebases

[8]. Paulina Hensman, David Masko

The Impact of Imbalanced Training Data for Convolutional Neural Networks

DEGREE PROJECT, IN COMPUTER SCIENCE

https://www.kth.se/social/files/588617ebf2765401cfcc478c/PHensmanDMasko_dkand15.pdf

[9] G. Weiss, Foster Provost,

Learning when training data are costly: the effect of class distribution on tree induction

Journal of Artificial Intelligence ResearchVol. 19, No. 1

https://dl.acm.org/doi/10.5555/1622434.1622445

References of used code bases, tutorials:

[A]https://www.kaggle.com/ky2019/starter-celebfaces-attributes-celeba-b5421ae1-e

[B] https://www.kaggle.com/saket0565/celebfaces-facial-attribute-recognition

[C] https://www.kaggle.com/bmarcos/image-recognition-gender-detection-inceptionv3

[D] https://www.kaggle.com/fkdplc/celeba-dcgan-for-generating-faces/notebook

[E] https://www.kaggle.com/nadergo/conditional-vae-on-faces

[F] https://www.kaggle.com/biswarupray/vae-celebface