Universität Augsburg
Fakultät für Angewandte
Informatik

Emotional face generation with adversarial networks

Nicolas Kolbenschlag

Supervisor: Vincent Karas, M.Sc.

Seminar Computational Intelligence, 23.07.2020

Agenda

About the topic

What, why and how?

- The successful training of machine learning models depends heavily on the quality and quantity of the used training data
- The goal of this work is to support the classification task of human emotion recognition, by **generating new images of human faces expressing specific emotions**
- Therefore, a Generative Adversarial Network (GAN) was trained on existing and annotated images from the AffWild2-Dataset [10]

Examples of the dataset (AffWild2)

Generative adversarial neural networks

What is a GAN?

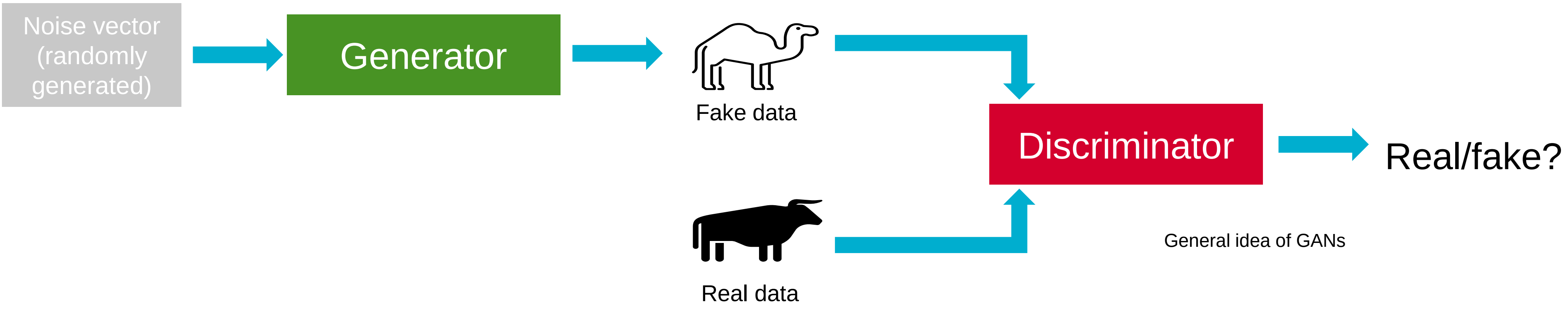
- Neural network architecture for unsupervised learning
- Name origin:
  - Generative: It generates new samples
  - Adversarial: That is done by “playing out two neural nets against each other”
- Consists of two different neural networks
  - Generator: Takes noise as input and outputs the “fake data” (data with same shape as the target samples)
  - Discriminator: Takes data (real and fake) as input and is trained to classify whether input is real (from dataset) or fake (from the generator)

“We propose a new framework for estimating generative models via an adversarial process, in which we simultaneously train two models: a generative model G that captures the data distribution, and a discriminative model D that estimates the probability that a sample came from the training data rather than G. The training procedure for G is to maximize the probability of D making a mistake.” [1]

Generative adversarial neural networks

What is a GAN?

- The aim is to confuse the discriminator, so that it can’t predict whether input is real or fake
- If that is achieved, the output of the generator could have been part of the original training set
- One training step of a GAN:
  1. Generate (predict) fake samples with generator
  2. Train **discriminator with these fakes and label FAKE** and some **real data and label REAL**
  3. Train the GAN with **discriminator weights frozen on label REAL** (the only way to minimize training error is that output of the generator is like real data)



General idea of GANs

Generating random faces

First approach

- Before trying to generate specific emotions in human faces, the goal was to get a basic GAN, like described before, working
- Therefore, the labels of the images from the dataset were completely ignored and a basic GAN was trained on ~3k images from ~200 different people

Generated images after 10 epochs

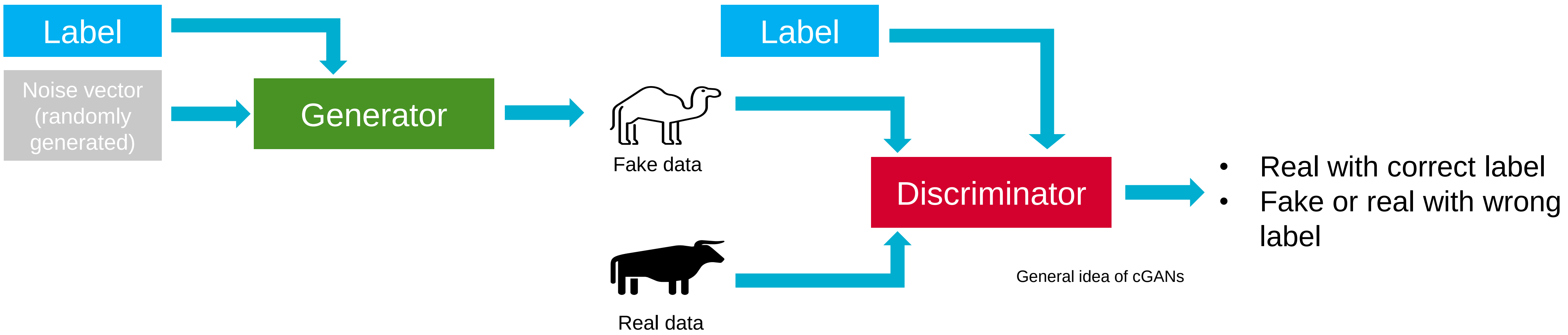
Generated images after 50 epochs

Generated images after 100 epochs

Generating specific emotions

What is a Conditional GAN (cGAN)?

- After generating faces without paying attention to their emotions, the labels of the dataset (condition) were added to the GAN
- A conditional GAN works like a usual unconditional GAN, but both the generator and the discriminator are given the label as additional input [2]
- The discriminator not only predicts whether the input image is real or fake, but rather if whether it is real with the correct emotion or fake or with the wrong emotion



General idea of cGANs

Difficulties encountered during training

Mode collapse and Wasserstein-loss

- If the generator finds a relatively good output, it may learn to only produce that output
- The discriminator now should/could reject it always, but if it is stuck into a local minimum, it doesn't and it is too easy for the generator, because he can output his same image again and again
- This error is called **mode collapse** [11]

Outputs of collapsed generator

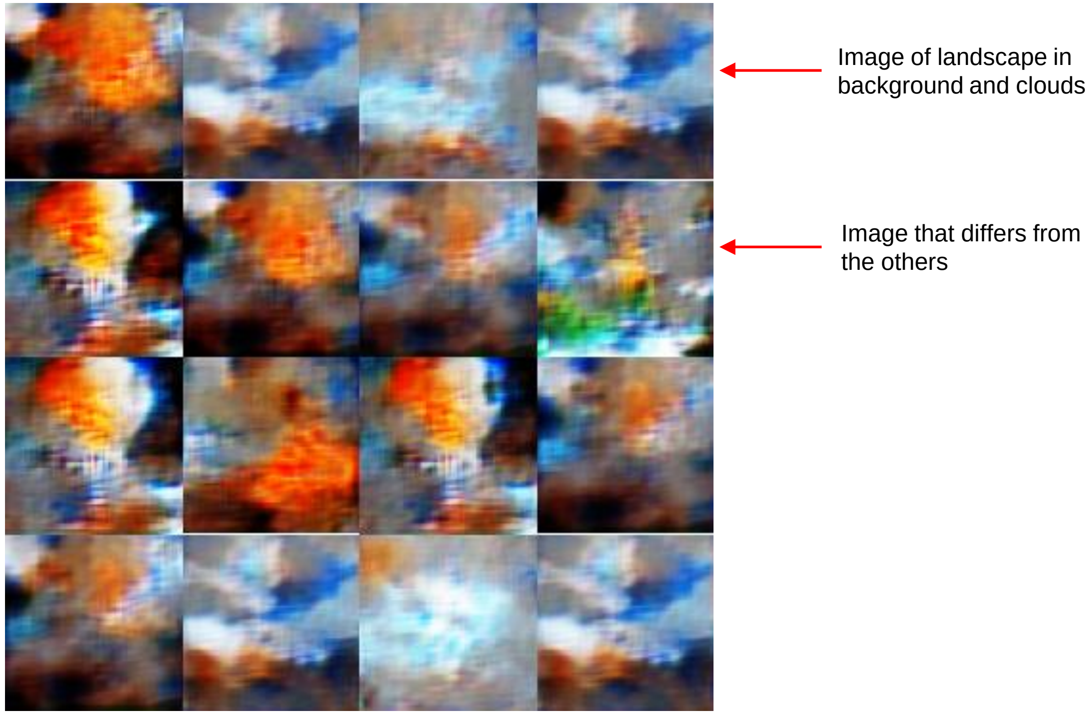
- An approach to fix it is the **Wasserstein-loss-function** [7]
- The discriminator with the Wasserstein-loss doesn't classify binary (real/fake), but it predicts the authenticity of a sample continuously (-∞, +∞) → tries to avoid vanishing gradients
- In principle it is very hard to say why a specific GAN collapses and how it can be fixed (even with Wasserstein-loss)

Difficulties encountered during training

Trying to overcome the collapse

- To rule out that a fundamental error in the model architecture causes the trouble, the model was trained on a different dataset (images of landscapes labelled according to the weather) [12]
- Although most generated images look similar, this experiment proved that the model could generate images without collapsing every time

Samples of the dataset



Generated images

Results and evaluation

The final results

- After varying the hyperparameters and model architecture several time, the model began to sometimes generate different images of faces
- Notable variation of the model architecture: the discriminator had two losses (Wasserstein for authenticity of the generated image and usual multi-classification for the category/emotion) [9]
- Although the generated images are not really usable, the results show the feasibility of generating human faces expressing emotions

Image that differs from the others

Generated images

Future work

What could happen next?

1. Improve results
   - More training: more data and more epochs
   - Tune hyperparameter and vary model architecture (find combination that doesn’t collapse)
2. Using for data augmentation
   - The initial idea for the project was to improve classification of emotional faces by providing additional training samples
   - A classifier could be trained on the real data and then on both real data and synthesized samples
   - If the second approach delivers better results, the project would have served its purpose

References

Papers and sample code

1. Goodfellow, Ian J., et al. „Generative Adversarial Networks“. arXiv:1406.2661 [cs, stat], June 2014. arXiv.org, http://arxiv.org/abs/1406.2661..
2. Mirza, Mehdi, and Simon Osindero. „Conditional Generative Adversarial Nets“. arXiv:1411.1784 [cs, stat], November 2014. arXiv.org, http://arxiv.org/abs/1411.1784.
3. Han, Jing, et al. „Adversarial Training in Affective Computing and Sentiment Analysis: Recent Advances and Perspectives [Review Article]“. IEEE Computational Intelligence Magazine, Bd. 14, Nr. 2, May 2019, S. 68–81. IEEE Xplore, doi:10.1109/MCI.2019.2901088.
4. Vielzeuf, Valentin, et al. „The Many Variations of Emotion“. 2019 14th IEEE International Conference on Automatic Face Gesture Recognition (FG 2019), 2019, S. 1–7. IEEE Xplore, doi:10.1109/FG.2019.8756560.
5. Atienza, Rowel. „GAN by Example Using Keras on Tensorflow Backend“. Medium, April 28th, 2017, https://towardsdatascience.com/gan-by-example-using-keras-on-tensorflow-backend-1a6d515a60d0.
6. Brownlee, Jason. „How to Develop a Conditional GAN (CGAN) From Scratch“. Machine Learning Mastery, July 4th, 2019, https://machinelearningmastery.com/how-to-develop-a-conditional-generative-adversarial-network-from-scratch/.
7. Arjovsky, Martin, et al. „Wasserstein GAN“. arXiv:1701.07875 [cs, stat], December 2017. arXiv.org, http://arxiv.org/abs/1701.07875.
8. Brownlee, Jason. „How to Develop a Wasserstein Generative Adversarial Network (WGAN) From Scratch“. Machine Learning Mastery, July 16th, 2019, https://machinelearningmastery.com/how-to-code-a-wasserstein-generative-adversarial-network-wgan-from-scratch/.
9. cnx. bobchennan/Wasserstein-GAN-Keras. 2017. 2020. GitHub, https://github.com/bobchennan/Wasserstein-GAN-Keras.

References

Papers and sample code

10. i·bug - resources - Aff-Wild2 database. https://ibug.doc.ic.ac.uk/resources/aff-wild2/.
    - Kollias, Dimitrios, and Stefanos Zafeiriou. „Expression, Affect, Action Unit Recognition: Aff-Wild2, Multi-Task Learning and ArcFace“. arXiv:1910.04855 [cs, eess], September 2019. arXiv.org, http://arxiv.org/abs/1910.04855.
    - Kollias, Dimitrios, and Stefanos Zafeiriou. „Aff-Wild2: Extending the Aff-Wild Database for Affect Recognition“. arXiv:1811.07770 [cs, stat], December 2019. arXiv.org, http://arxiv.org/abs/1811.07770.
    - Kollias, Dimitrios, and Stefanos Zafeiriou. „A Multi-Task Learning & Generation Framework: Valence-Arousal, Action Units & Primary Expressions“. arXiv:1811.07771 [cs, stat], December 2019. arXiv.org, http://arxiv.org/abs/1811.07771.
    - Kollias, Dimitrios, et al. „Deep Affect Prediction in-the-wild: Aff-Wild Database and Challenge, Deep Architectures, and Beyond“. International Journal of Computer Vision, Bd. 127, Nr. 6–7, June 2019, S. 907–29. arXiv.org, doi:10.1007/s11263-019-01158-4.
    - Zafeiriou, Stefanos, et al. „Aff-Wild: Valence and Arousal ‘In-the-Wild’ Challenge“. 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE, 2017, S. 1980–87. DOI.org (Crossref), doi:10.1109/CVPRW.2017.248.
    - Kollias, Dimitrios, et al. „Recognition of Affect in the Wild Using Deep Neural Networks“. 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE, 2017, S. 1972–79. DOI.org (Crossref), doi:10.1109/CVPRW.2017.247.
11. „Common Problems | Generative Adversarial Networks“. Google Developers, https://developers.google.com/machine-learning/gan/problems?hl=de.
12. Gbeminiyi Oluwafemi, AJAYI, und WANG Zenghui. „Multi-Class Weather Classification from Still Image Using Said Ensemble Method“. 2019 Southern African Universities Power Engineering Conference/Robotics and Mechatronics/Pattern Recognition Association of South Africa (SAUPEC/RobMech/PRASA), 2019, S. 135–40. IEEE Xplore, doi:10.1109/RoboMech.2019.8704783.

Thank you for your attention.

Do you have any questions?

Universität Augsburg
Fakultät für Angewandte
Informatik

Nicolas Kolbenschlag

Universität Augsburg

nicolas.kolbenschlag@student.uni-augsburg.de

www.uni-augsburg.de