Deepfaking and how it works

By Fahim Tanvir

What is deepfaking?

Deepfaking, is all about manipulating audio or imagery to realistically imitate someone’s speech or facial movements through training and testing models using artificial intelligence and deep neural networks. I

This became popular during 2020 but was around at least since the 2010s. This has created trends such as internet humor regarding impersonating celebrities(for example, there are countless memes of people using ai to mimic the US presidents) to other, more serious matters.

Why is it relevant?

It’s relevance, especially as of modern history of the 2020s, is due to negative and positive aspects relating to it.

For example, a negative aspect of deepfaking is of course, it has been used for illegal impersonation and fraud, which adds controversy and concern over it.

Tom Hanks says AI version of him used in dental plan ad without his consent | Tom Hanks | The Guardian

(Summary: Tom Hanks was imprisoned by someone using a deepfaked video of him for advertisement)

However some good does come out of it as it has some uses:

AI is taking over the iconic voice of Darth Vader, with the blessing of James Earl Jones | TechCrunch

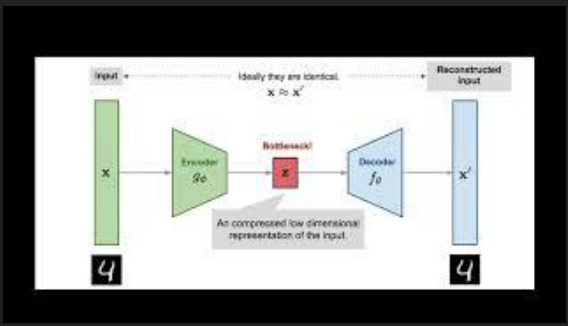
(Summary: Disney using deepfaking for voice acting as the actor for Darth Vader, James Earl Jones is aging. Article uses another example where it was used for voice lines for another actor of another film(Val Kilmer) due deal with his sickness during production(Coldewey 5)

How does it work?

Basically, deep-faking works through an algorithm which process an output, transposes it on the model and reconstructs it for the out.

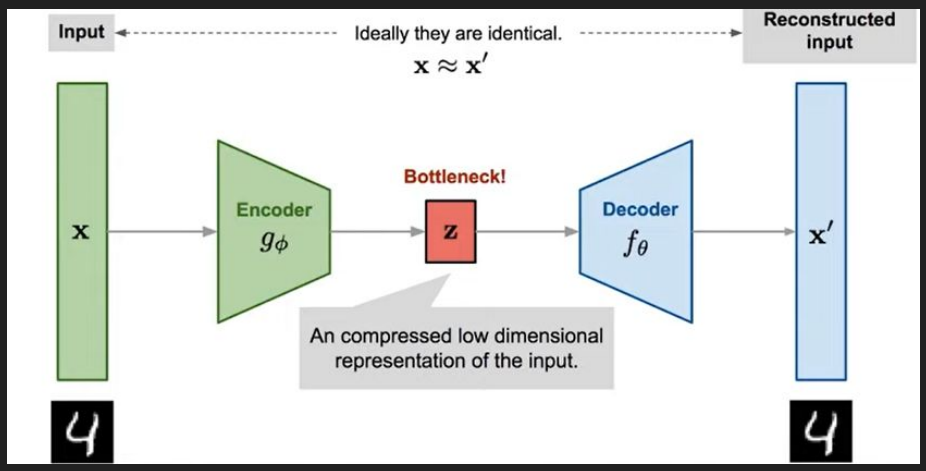
This video by Ian Sullivan goes over it in more detail:

https://www.youtube.com/watch?v=XqluthtTenI&t=0s



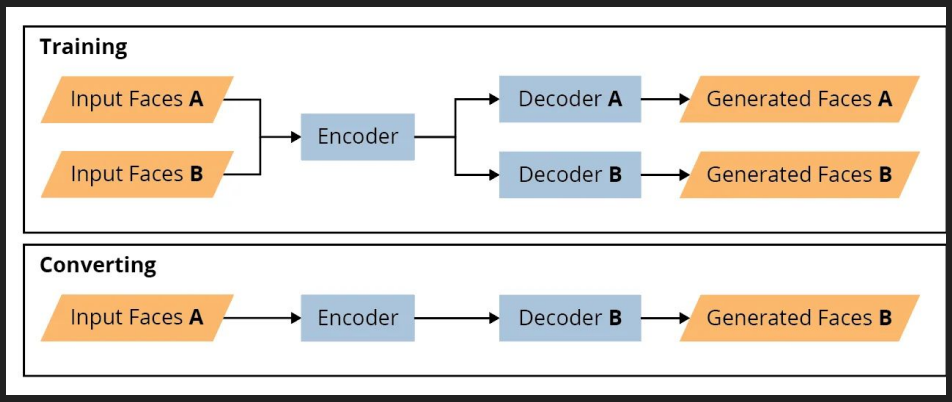
But to summarize his video, the DNN that creates deepfakes uses an autoencoder, which is comprised of 3 parts: an encoder part, bottleneck/training part and a decoder part.

How does it work?(cont'd)



(top)Screenshot from last slide's video video(0:42-3:00)

(bottom)Diagram from Carnegie Mellon article above paragraph 14



Terminology(just for clarity)

Model:Video or audio file user wants to replicate or mimic

Input: Either an image of someone's face or someone's voice file

Output:The end result, being a video or sound file of the input mimicking the model

According to "How Easy Is It to Make and Detect a Deep Fake?" by Carnegie Mellon students Catherine Bernaciak and Dominic Ross, the program used tracks the facial features of the person being used for the model and the input through a Deep Neural Network(Bernaciak and Ross 14). That's the encoder part Sullivan described as the data becomes compressed into training data.

Speaking of which, next comes training or the bottleneck/training step, which is where the DNN trains using the input data as "then each passed separately through decoder networks for the A and B faces that attempt to generate, or recreate, each set of faces separately"(Bernaciak and Ross 15). In context with image deepfaking, it detects any facial features and movements of a video's frames, and then recreates it with the input image. Essentially, the program being used compares the faces of the model's frames and the input back and forth as it is being transposed.

Finally, there's the docoder, which is described as "When the output of the encoder is passed to the decoder for B, it will attempt to generate face B swapped with the identity of A"(Bernaciak and Ross 16). Essentially the model's facial positions become switched with the inputs. New frames are produced and compiled, creating a deepfaked video.

How does it work(cont’d)

For audio, it works the same way, just replace the input with a song instead of an image of a face and a model with another audio file that will be mimicked by the input. It takes every sound byte of an input (in this case a sound file) and transposes it on the model(the sound file whose speech the use would would want to imitate). For example, if one would want their voice to sing a song without singing, the deep-fake ai or program would generate the sound by transposing the entire sound file of them just speaking on a recording of a song.

Demonstration:

Lots of programs and applications have been made by people(usually by python as that coding language is used extensively for artificial intelligence). It is very easy to find many on github. These 2 are just examples I found and have knowledge of prior

Program 1(Image deepfaking)

https://colab.research.google.com/github/AliaksandrSiarohin/first-order-model/blob/master/demo.ipynb#scrollTo=Oxi6-riLOgnm

Program 2(Audio deepfaking)

https://colab.research.google.com/drive/1r4IRL0UA7JEoZ0ZK8PKfMyTIBHKpyhcw#scrollTo=3KyMRbK49g__

Looking inside program 1:

For Aliaksandr Siarohin's program, most of the deepfake relies on the function "generate".

```
def generate(button):
  main.layout.display = 'none'
  loading.layout.display = ''
  filename = model.value + ('' if model.value == 'fashion' else '-cpk') + '.pth.tar'
  if not os.path.isfile(filename):
    response = requests.get('https://github.com/graphemecluster/first-order-model-demo/releases/download/checkpoints/' + filename, stream=True)
    with progress_bar:
      with tqdm.wrapattr(response.raw, 'read', total=int(response.headers.get('Content-Length', 0)), unit='B', unit_scale=True, unit_divisor=1024) as raw:
        with open(filename, 'wb') as file:
          copyfileobj(raw, file)
    progress_bar.clear_output()
  reader = imageio.get_reader(selected_video, mode='I', format='FFMPEG')
  fps = reader.get_meta_data()['fps']
  driving_video = []
  for frame in reader:
    driving_video.append(frame)
  generator, kp_detector = load_checkpoints(config_path='config/%s-256.yaml' % model.value, checkpoint_path=filename)
  with progress_bar:
    predictions = make_animation(
      skimage.transform.resize(numpy.asarray(selected_image), (256, 256)),
      [skimage.transform.resize(frame, (256, 256)) for frame in driving_video],
      generator,
      kp_detector,
      relative=relative.value,
      adapt_movement_scale=adapt_movement_scale.value
```

(Red)"Reader" Takes all the frames of the video and the input images.
(Yellow) Checkpoints or pinpoints trains the program to detect facial features(such as eyes, nose, mouth) of both the video and image and transposes the latter with the former.
(Green) "Predictions" create new frames using the input which was transposed with the video, producing the output video.

Looking inside program 2:

The sound program uses tensorflow to transpose and sift through soundbytes of the 2 sound files. It also uses google drive to store data unlike the previous one. I will go over it more directly during presentation .

CONCLUSION

Those 2 were just examples. Many people have created similar programs to do the same thing as the base structure(the autoencoder inside a DNN) is the same or atleast the idea of it. Deepfaking, just like anything trendy involving artificial intelligence, is constantly evolving as more people are finding out and tampering with it. So it definitely isn’t going away and its uses, whether it’s for bad(aforementioned fraud) or good(convenient for voice actors and preservation) all stems from the user’s mentality. With that said, it is very interesting and can be useful to anyone who wants to learn about artificial intelligence or even just coding on python.