REPORT
Text to Image/Video synthesis using GANs

Approach:

The objectives of the project can be divided into two sections.

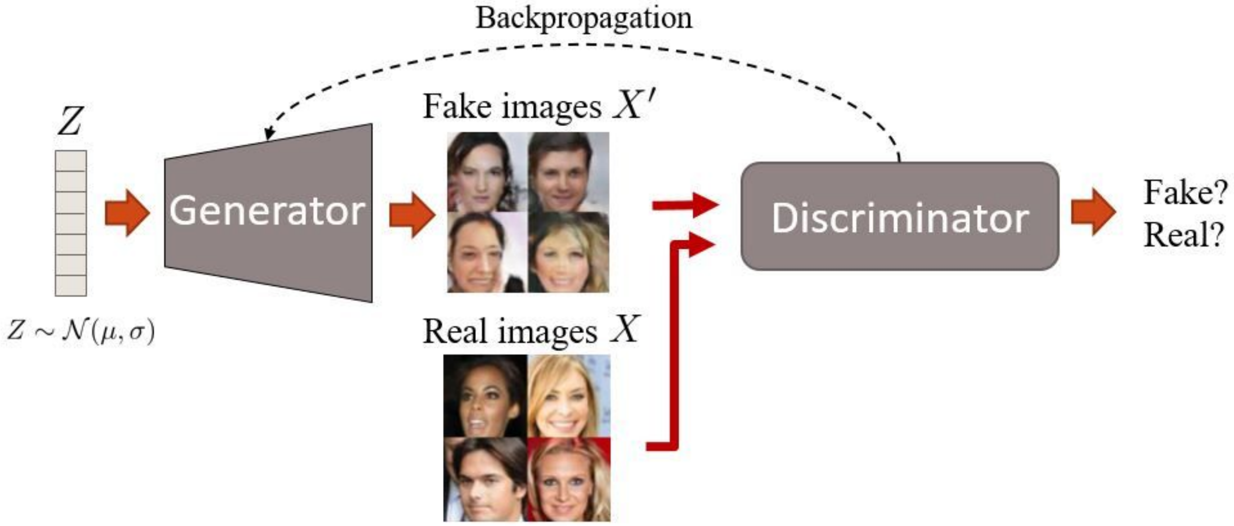
Firstly to generate Images from Text data we will use a deep convolutional generative adversarial network (DC-GAN) conditioned on text features encoded by a hybrid character-level convolutional recurrent neural network]. This network can be trained on the ms coco dataset or any other dataset . Once this network has generated images from the corresponding text we will send it to our video generator.

Our video generation will network will do the job of generating a short video from this image. This network can be a kind of LSTM which will predict the next frame and from that frame predict the next frame or a GAN can also be employed which will employ a spatio-temporal convolutional architecture.The video generator would need to be trained on a large unlabeled video dataset.

Different Algorithms suitable for the task of generating images from text:

1.GAN(generative adversarial network)

In this adversarial nets framework, the generative model is pitted against an adversary: a discriminative model that learns to determine whether a sample is from the model distribution or the data distribution. The generative model can be thought of as analogous to a team of counterfeiters, trying to produce fake currency and use it without detection, while the discriminative model is analogous to the police, trying to detect the counterfeit currency. Competition in this game drives both teams to improve their methods until the counterfeits are indistinguishable from the genuine articles.In this article, we explore the special case when the generative model generates samples by passing random noise through a multilayer perceptron, and the discriminative model is also a multilayer perceptron. We refer to this special case as adversarial nets. In this case, we can train both models using only the highly successful backpropagation and dropout algorithms and sample from the generative model using only forward propagation



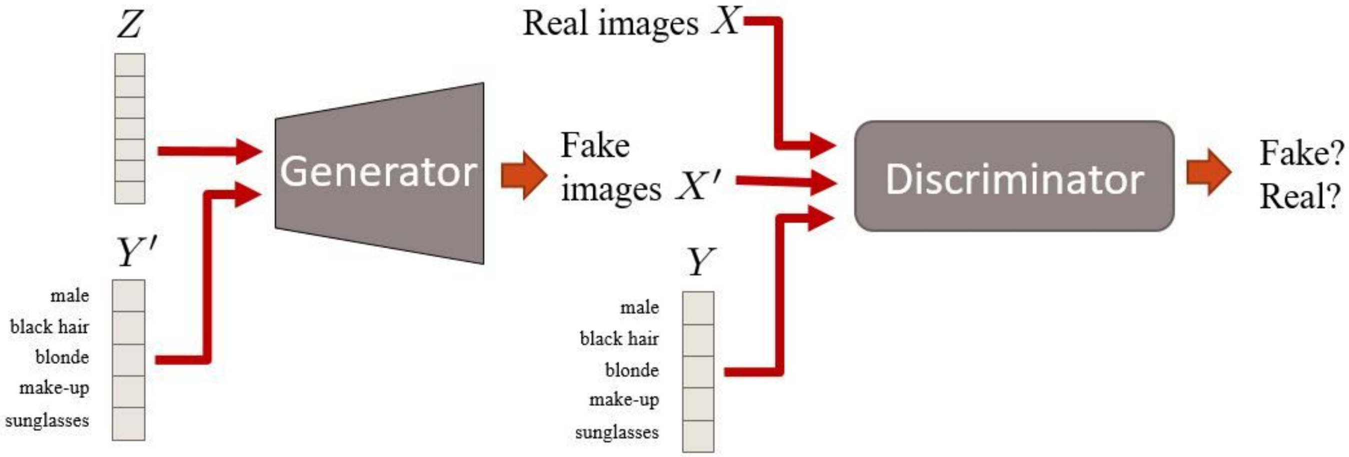
2. DCGAN(deep convolutional generative adversarial network)

In recent years, supervised learning with convolutional networks (CNNs) has seen huge adoption in computer vision applications. Comparatively, unsupervised learning with CNNs has received less attention. This method bridges the gap between the success of CNNs for supervised learning and unsupervised learning.this introducse a class of CNNs called deep convolutional generative adversarial networks (DCGANs), that have certain architectural constraints, and demonstrate that they are a strong candidate for unsupervised learning.DCGANs avoid using pooling and just use strided convolutions .They have introduced batch normalisation in both the networks.DCGANs avoid using fully connected networks.Deep convolutional gans improve upon the training stability and generate higher quality images.

3.CGAN(conditional generative adversarial network)

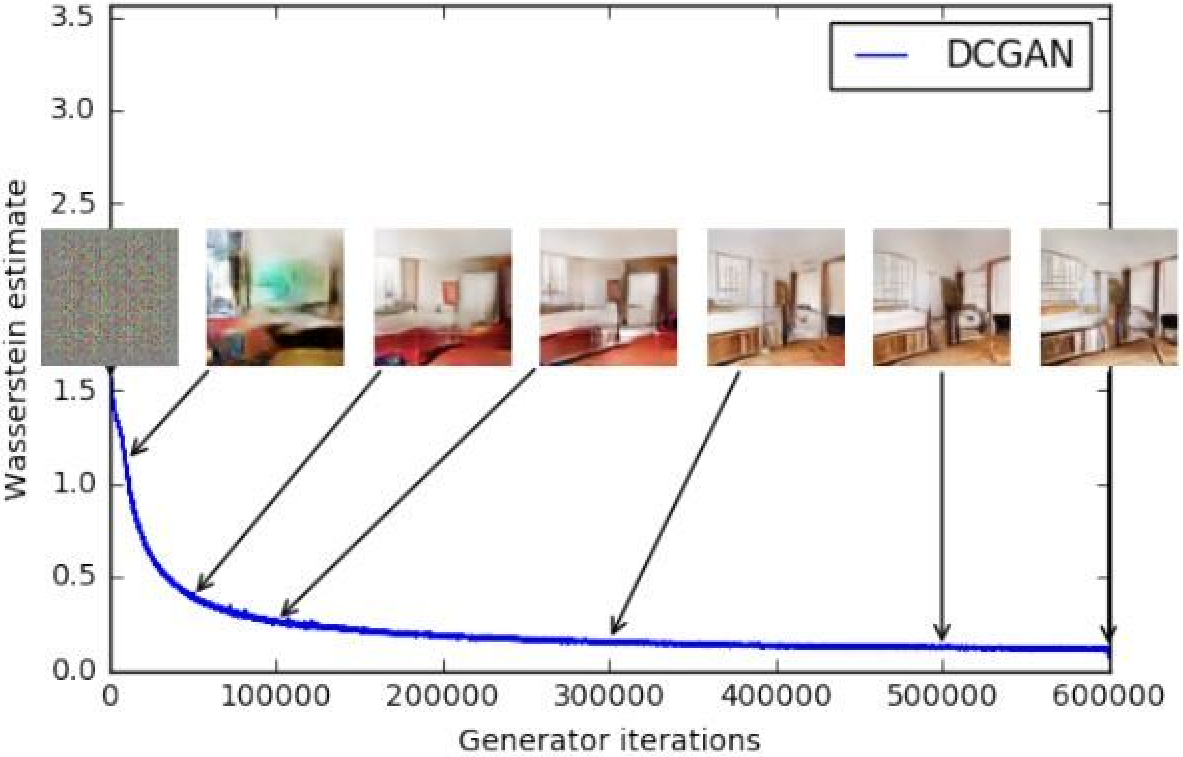
In an unconditioned generative model, there is no control on modes of the data being generated. However, by conditioning the model on additional information it is possible to direct the data generation process. Such conditioning could be based on class labels, on some part of data for inpainting like , or even on data from different modalities.Generative adversarial nets can be extended to a conditional model if both the generator and discriminator are conditioned on some extra information y. y could be any kind of auxiliary information, such as class labels or data from other modalities. We can perform the conditioning by feeding y into the both the discriminator and generator as additional input layer.

conditional GANs use extra label information which results in better quality images and being able to control – to an extent – how generated images will look



4.WGAN(warsesstien generative adversarial network)

GANs can produce very visually appealing samples, but are often hard to train, and much of the recent work on the subject has been devoted to finding ways of stabilizing training. Despite this, consistently stable training of GANs remains an open problem.Their proposed alternative, named Wasserstein GAN (WGAN), leverages the Wasserstein distance to produce a value function which has better theoretical properties than the original. WGAN requires that the discriminator (called the critic in that work) must lie within the space of 1-Lipschitz functions, which the authors enforce through weight clipping.WGAN demonstrates stable training of varied GAN architectures, performance improvements overweight clipping, high-quality image generation, and a character-level GAN language model without any discrete sampling. In this GAN we Change the loss function to include the Wasserstein distance. As a result, WGAN have loss functions that correlate with image quality. Also, training stability improves and is not as dependent on the architecture.With a meaningful loss function we can also tell when to stop the training.



We can see that the loss now correlates with actual image quality

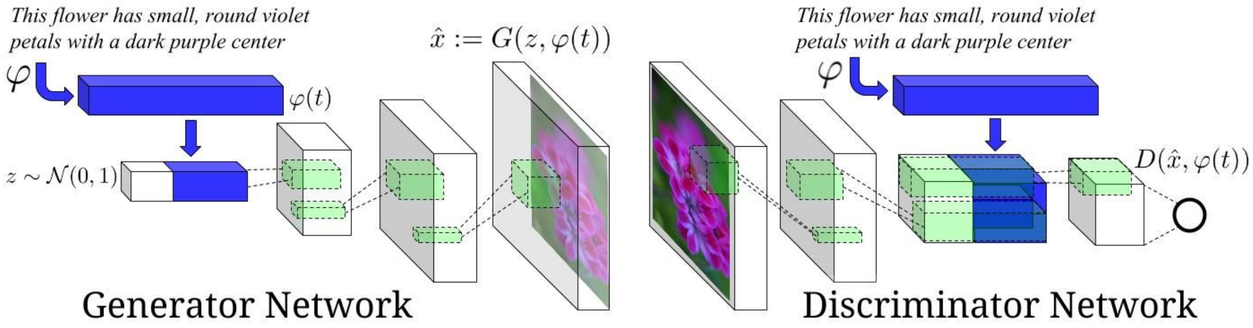
Our approach:

We will use a deep convolutional generative adversarial network (DC-GAN)[1] along with a hybrid character-level convolutional recurrent neural network.

The generator network is denoted $G : R^Z \times R^T \rightarrow R^D$, the discriminator as $D : R^D \times R^T \rightarrow \{0, 1\}$, where T is the dimension of the text description embedding, D is the dimension of the image, and Z is the dimension of the noise input to G (generator).

In the generator G, first we sample from the noise prior $z \in R^Z \sim N(0, 1)$ and we encode the text query t using text encoder ϕ. The description embedding $\phi(t)$ is first compressed using a fully-connected layer to a small dimension (in practice we used 128) followed by leaky-ReLU and then concatenated to the noise vector z. Following this, inference proceeds as in a normal deconvolutional network: we feed-forward it through the generator G; a synthetic image x` is generated via x`← $G(z, \phi(t))$. Image generation corresponds to feed-forward inference in the generator G conditioned on query text and a noise sample.

In the discriminator D, we perform several layers of strided convolution with spatial batch normalization followed by leaky ReLU. We again reduce the dimensionality of the description embedding $\phi(t)$ in a (separate) fully-connected layer followed by rectification. When the spatial dimension of the discriminator is 4 × 4, we replicate the description embedding spatially and perform a depth concatenation. We then perform a 1 × 1 convolution followed by rectification and a 4 × 4 convolution to compute the final score from D. Batch normalization is performed on all convolutional layers.


This flower has small, round violet petals with a dark purple center
φ
$\varphi(t)$
$z \sim \mathcal{N}(0,1)$
$\hat{x} := G(z, \varphi(t))$
Generator Network
This flower has small, round violet petals with a dark purple center
φ
$D(\hat{x}, \varphi(t))$
Discriminator Network

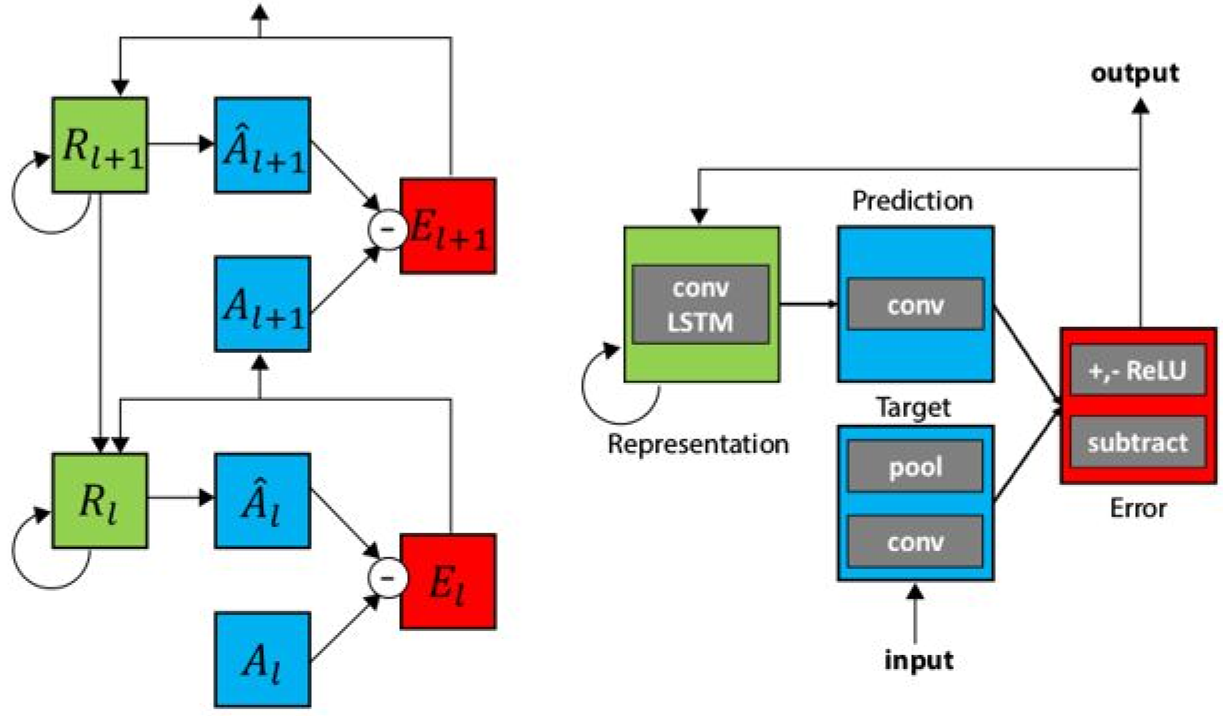
To get the text features from the text encoder, we first pre-train a deep convolutional recurrent text encoder on structured joint embedding of text captions with 1,024-dimensional GoogLeNet image embeddings.we used a hybrid of character-level ConvNet with a recurrent neural network (char-CNN-RNN). Pretraining is not an actual requirement of this network but we will apply pretraining to get quicker results and efficient experimentation.

The image that has been generated from this network will be used as input to anyone of the following video generation algorithms.

Different algorithms suitable for the task of generating videos from images:

1.Prednet, Deep predictive coding networks for video prediction and unsupervised learning.

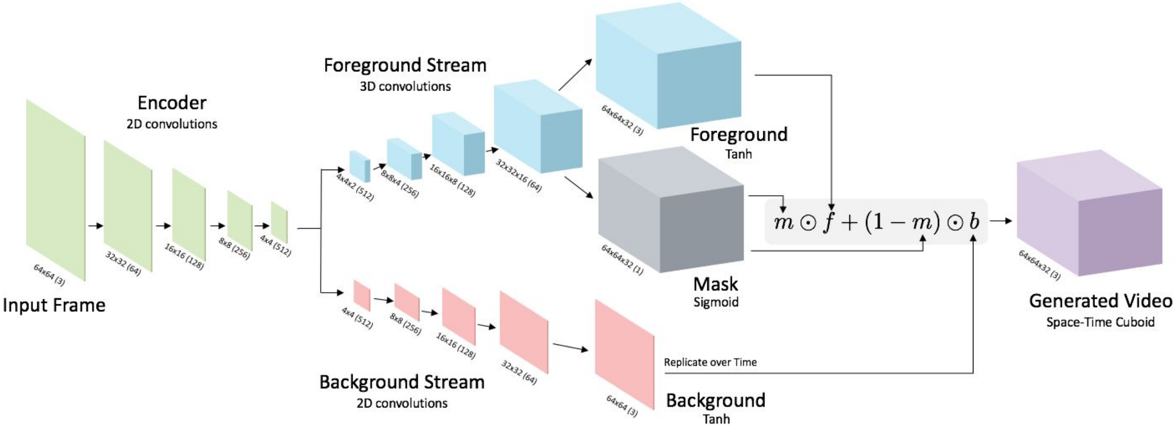
The PredNet architecture is diagrammed in the figure below. The network consists of a series of repeating stacked modules that attempt to make local predictions of the input to the module, which is then subtracted from the actual input and passed along to the next layer. Briefly, each module of the network consists of four basic parts: an input convolutional layer (A_L), a recurrent representation layer (R_L), a prediction layer (Aˆ_L), and an error representation (E_L). The representation layer, R_L , is a recurrent convolutional network that generates a prediction, Aˆ_L , of what the layer input, A_L , will be on the next frame. The network takes the difference between A^l and $A^{\text{ˆ}l}$ and outputs an error representation, E_L , which is split into separate rectified positive and negative error populations. The error, E_L , is then passed forward through a convolutional layer to become the input to the next layer (A_L+1). The recurrent prediction layer Rl receives a copy of the error signal E_L , along with top-down input from the representation layer of the next level of the network (R_L+1). The organization of the network is such that on the first time step of operation, the “right” side of the network (A_L’s and E_L’s) is equivalent to a standard deep convolutional network. Meanwhile, the “left” side of the network (the R_L’s) is equivalent to a generative deconvolutional network with local recurrence at each stage.



2.Video generation using a GAN that untangles the scene's foreground from the background.

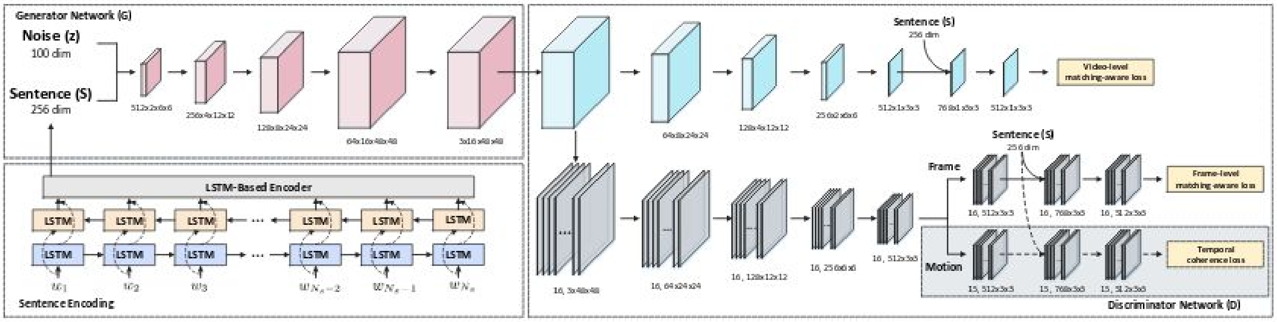
This builds on generative image models that leverage adversarial learning, which we apply to video. The basic idea behind the approach is to compete two deep networks against each other. One network ("the generator") tries to generate a synthetic video, and another network ("the discriminator") tries to discriminate synthetic versus real videos. The generator is trained to fool the discriminator.

For the generator, we use a deep convolutional network that inputs low-dimensional random noise and outputs a video. To model video, we use spatiotemporal up-convolutions (2D for space, 1D for time). The generator also models the background separately from the foreground. The network produces a static background (which is replicated over time) and a moving foreground that is combined using a maskWe simultaneously train a discriminator network to distinguish real videos from fake videos. We use a deep spatiotemporal convolutional network for the discriminator, and to generate future frames from a single image we will attach an encoder to the network . We illustrate this below:



3. Video synthesis from caption using TGAN-C

Architecture of TGAN is as in figure below



Bidirectional LSTM is used to contextually embed each word and encode word sequence into the sentence representation S.

Generator Network: Given the input sequence S and random noise variable z, a generator network is devised. The generator network first encapsulates random noise z and input sentence S into a fixed length input latent variable p, on which feature transformation and concatenation is applied and corresponding video is generated (through 3D deconvolution layers).

Discriminator Network: The discriminator network is designed for following three functions:

(1) distinguishing real video from synthetic one and aligning video with the correct caption (Video discriminator),

(2) determining whether each frame is real/fake and semantically matched/mismatched with the conditioning caption(Frame discriminator),

(3) exploiting the temporal coherence across consecutive real frames(Motion discriminator).

Losses:

Video-level matching-aware loss:

By minimising this loss , the video discriminator is trained to not only recognise each real video from synthetic ones but also classify semantically matched video-caption pair from mismatched ones.

Frame-level matching-aware loss:

Minimising this loss enforces the frame discriminator to discriminate whether each frame of the input video both real and semantically matched with the caption.

Temporal coherence loss:

Temporal coherence constraint loss, for generator matrix to create temporally coherent frames.

Temporal coherence adversarial loss: By minimizing the temporal coherence, the temporal discriminator is trained to not only recognize the temporal dynamics across synthetic frames from real ones but also align the temporal dynamics with the matched caption

References

Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, Honglak Lee -Generative Adversarial Text to Image Synthesis
https://arxiv.org/pdf/1605.05396.pdf

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. In NIPS, 2014.
https://arxiv.org/pdf/1406.2661

Mirza, M. and Osindero, S. Conditional generative adversarial nets
https://arxiv.org/pdf/1411.1784

Radford, A., Metz, L., and Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. 2016.
https://arxiv.org/pdf/1511.06434.pdf

Reed, S., Akata, Z., Lee, H., and Schiele, B. Learning deep representations for fine-grained visual descriptions. In CVPR, 2016.
https://arxiv.org/pdf/1605.05395

Ishaan Gulrajani , Faruk Ahmed, Martin Arjovsky , Vincent Dumoulin, Aaron Courville. Improved Training of Wasserstein GANs
https://arxiv.org/pdf/1704.00028.pdf

William Lotter, Gabriel Kreiman & David Cox.Deep Predictive Coding Networks For video prediction and unsupervised learning.
https://arxiv.org/pdf/1605.08104

Yingwei Pan, Zhaofan Qiu, Ting Yao, Houqiang Li and Tao Mei.Generating Videos from Captions
https://www.microsoft.com/en-us/research/wp-content/uploads/2017/11/BNI02-panA.pdf