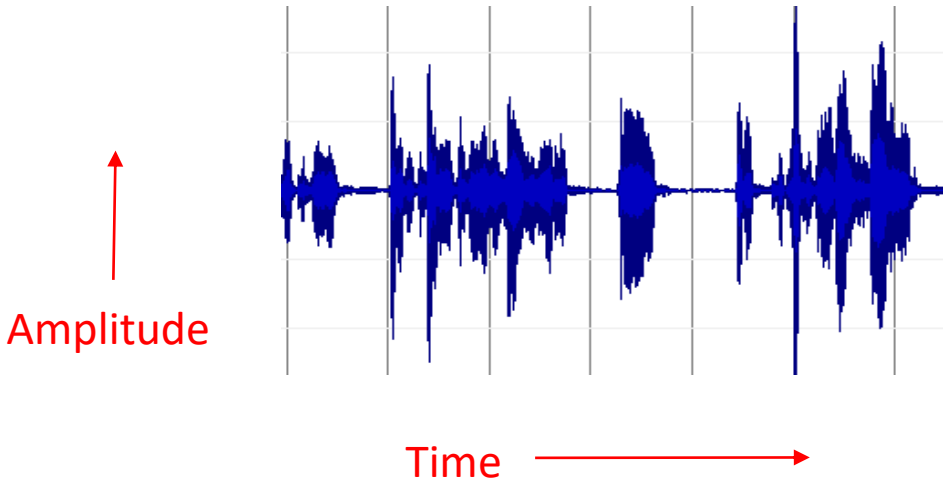
Discussion 11

EE599 Deep Learning

Arindam Jati, Arnab Sanyal

Spring 2020

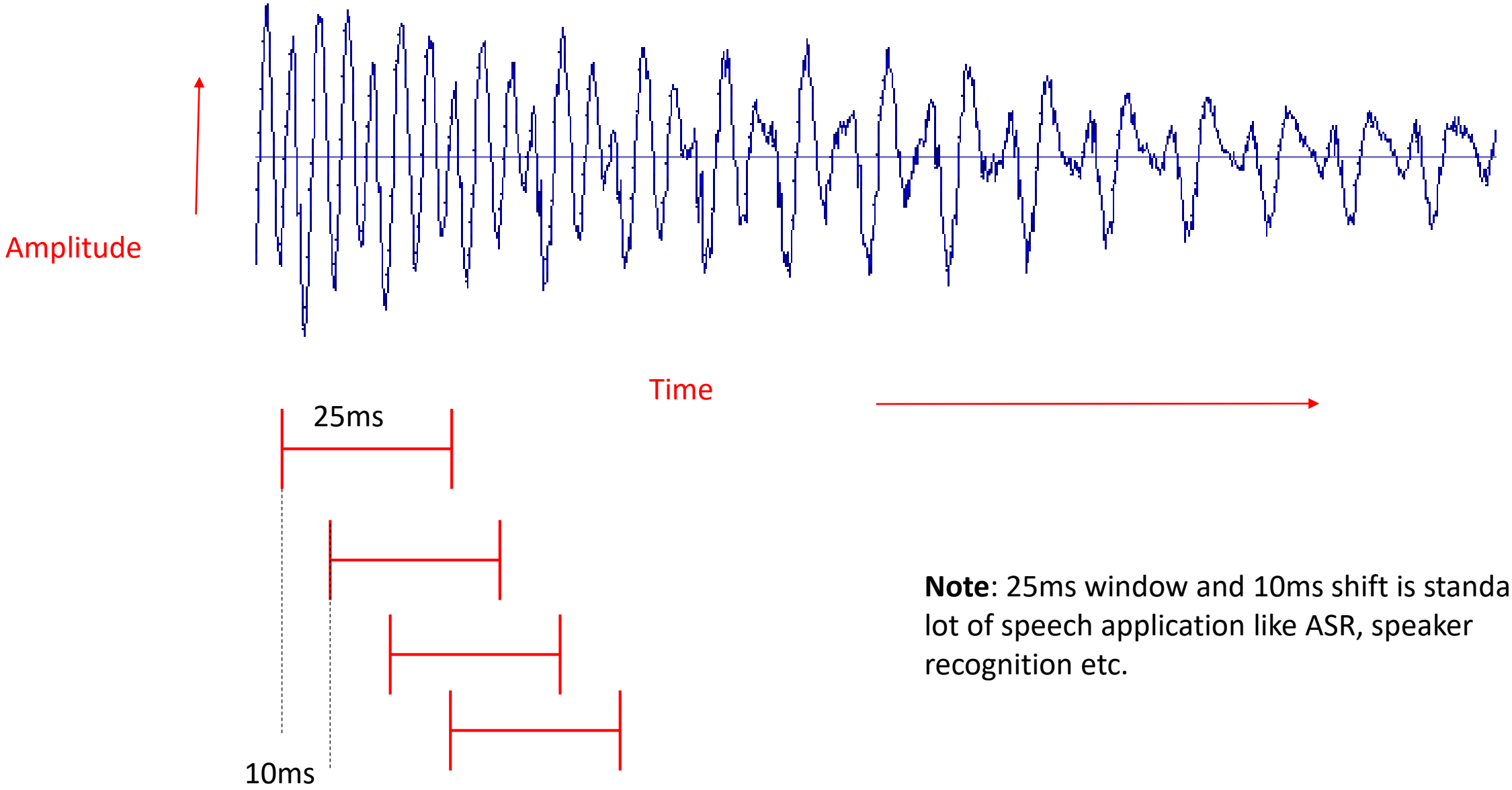
Audio signal representation



Recap from DSP discussion:

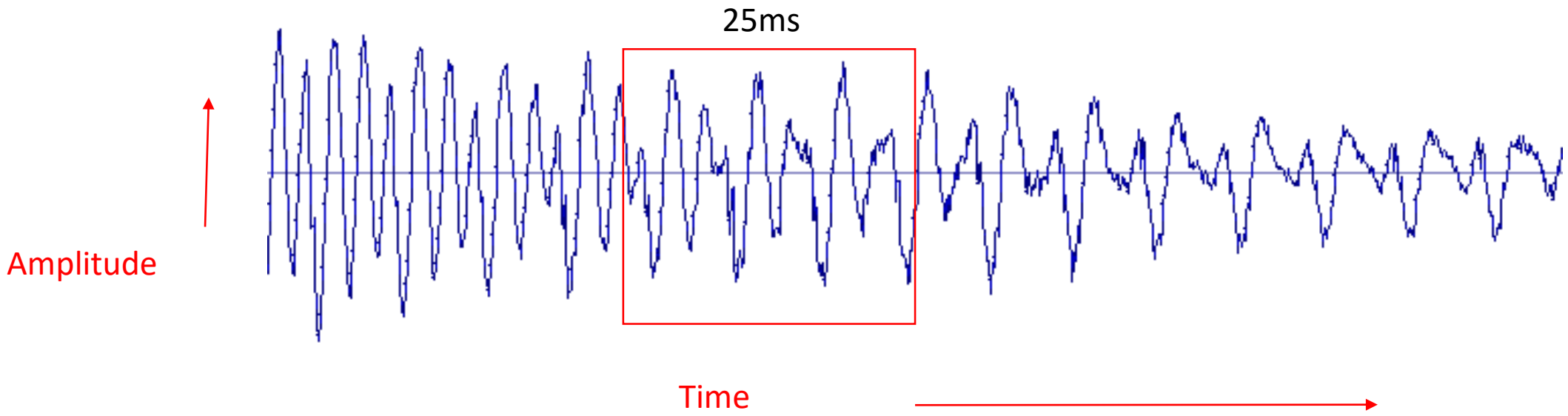
- Why don't we use the time domain signal directly?

- We want to go to frequency domain
- Problem: Audio signal is *not* stationary
- Solution: Short-time stationarity assumption



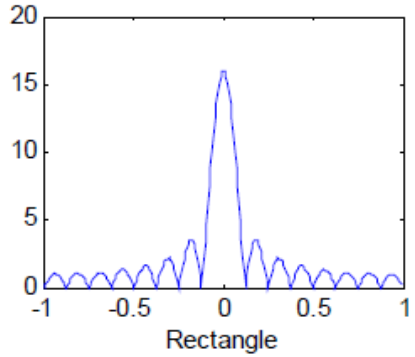
Note: 25ms window and 10ms shift is standard in a lot of speech application like ASR, speaker recognition etc.

Windowing



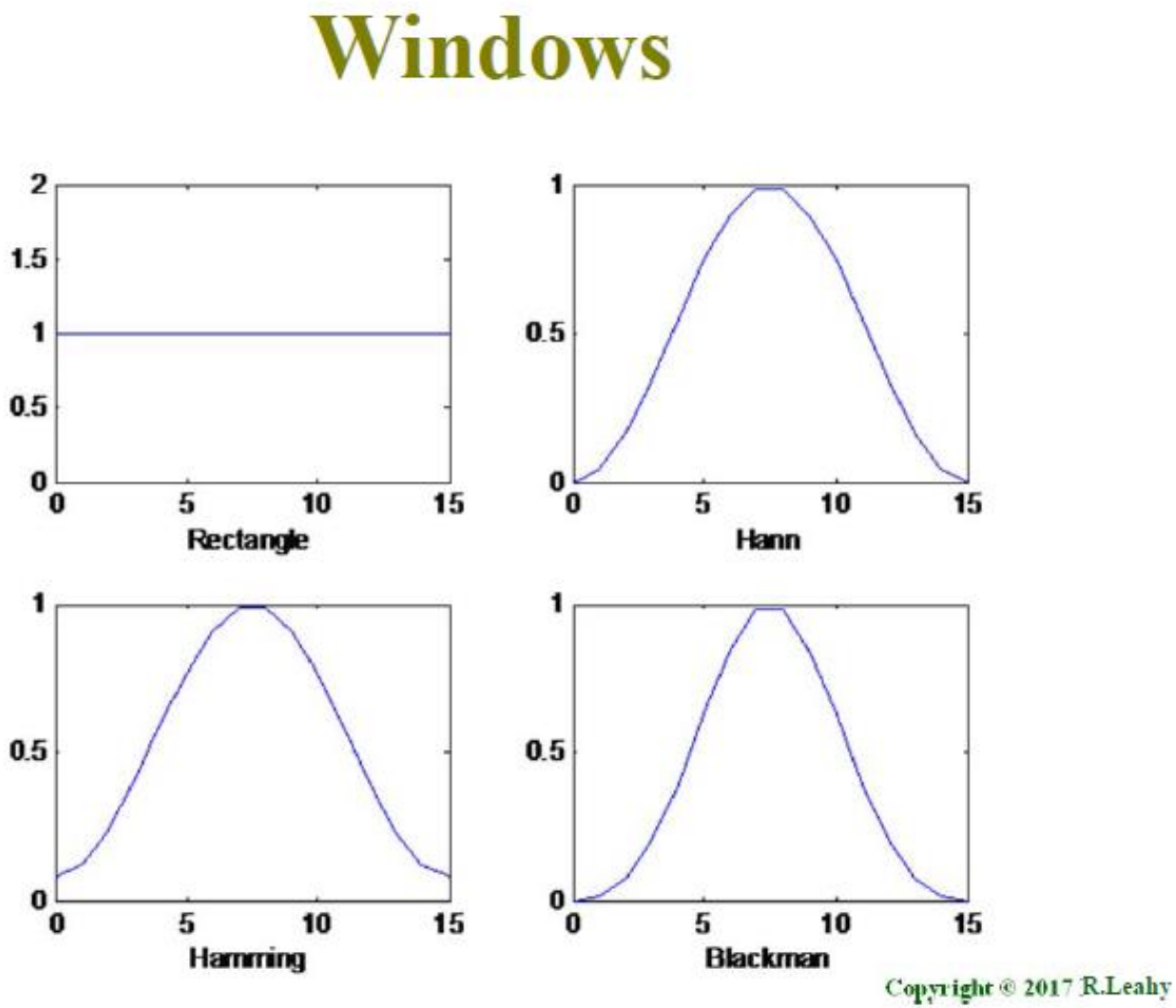
Problems with rectangular Windows

DTFT



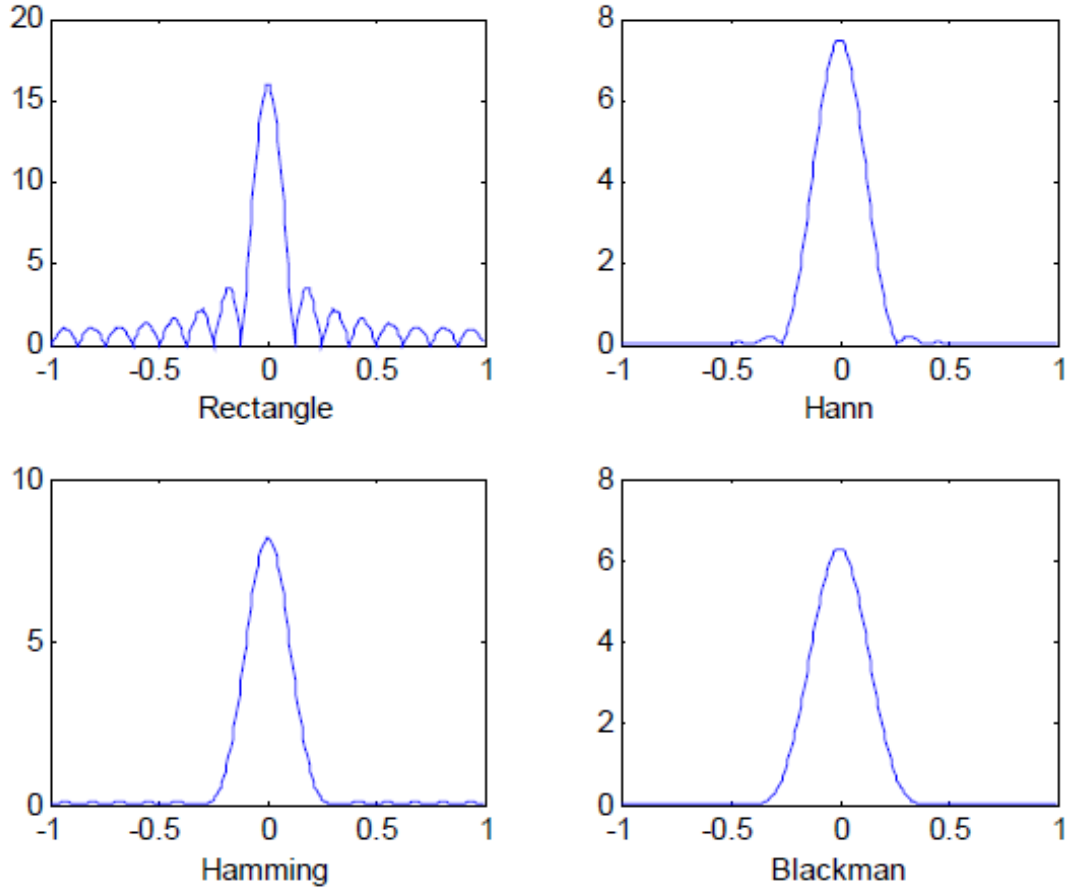
Higher
Sidelobe
Leakage

Windowing – EE 483 Recap



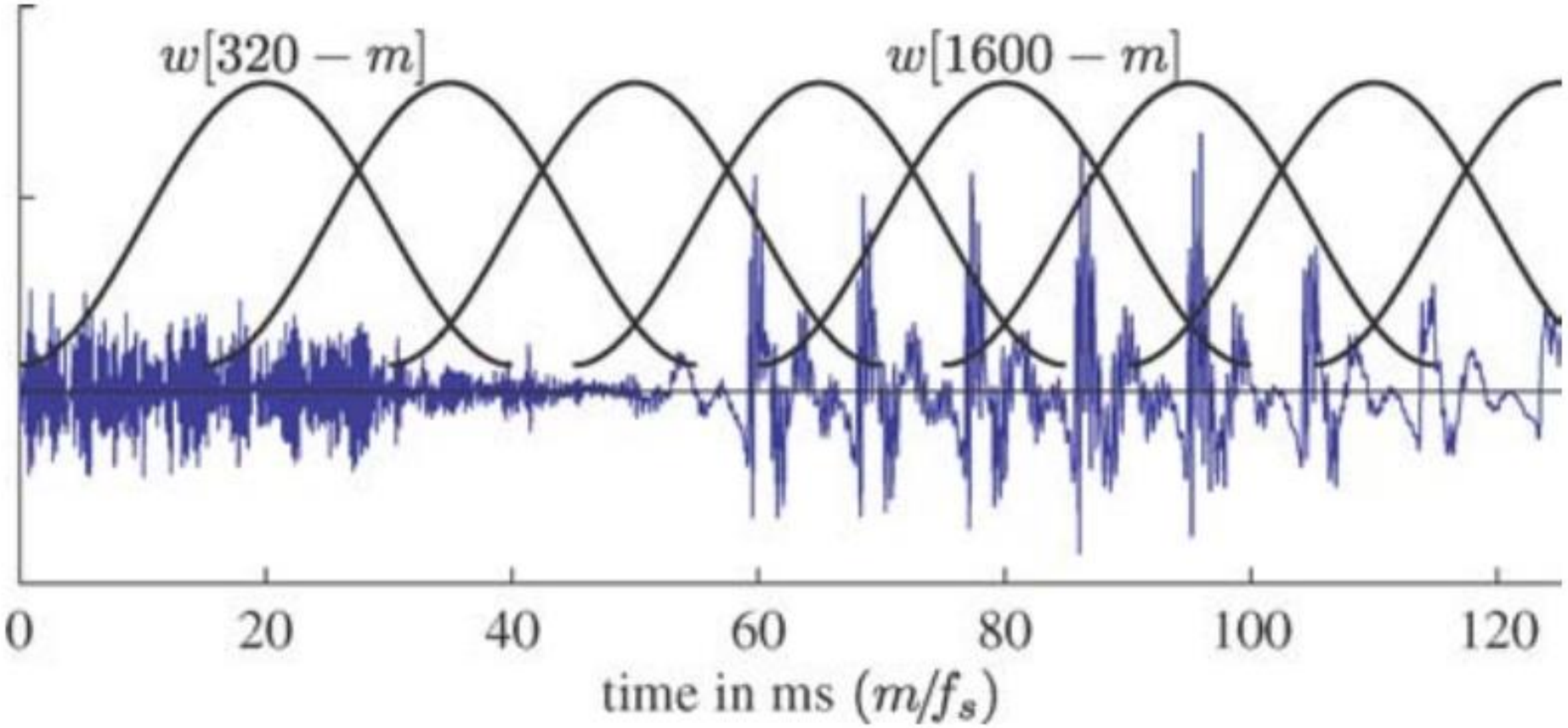
Windowing – EE 483 Recap

Windows: DTFTs



Hann, Hamming and Blackman windows have lower sidelobe energy

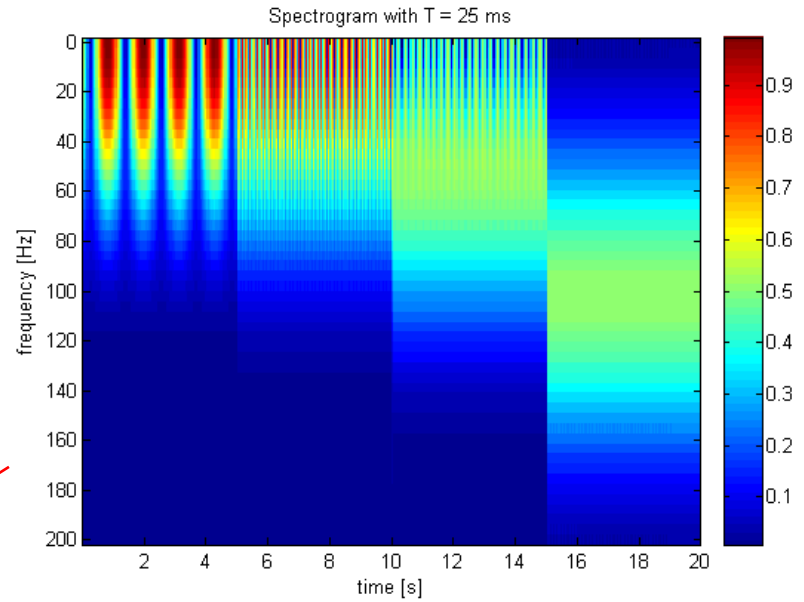
Applying a Hamming Window



Short-time Fourier Transform

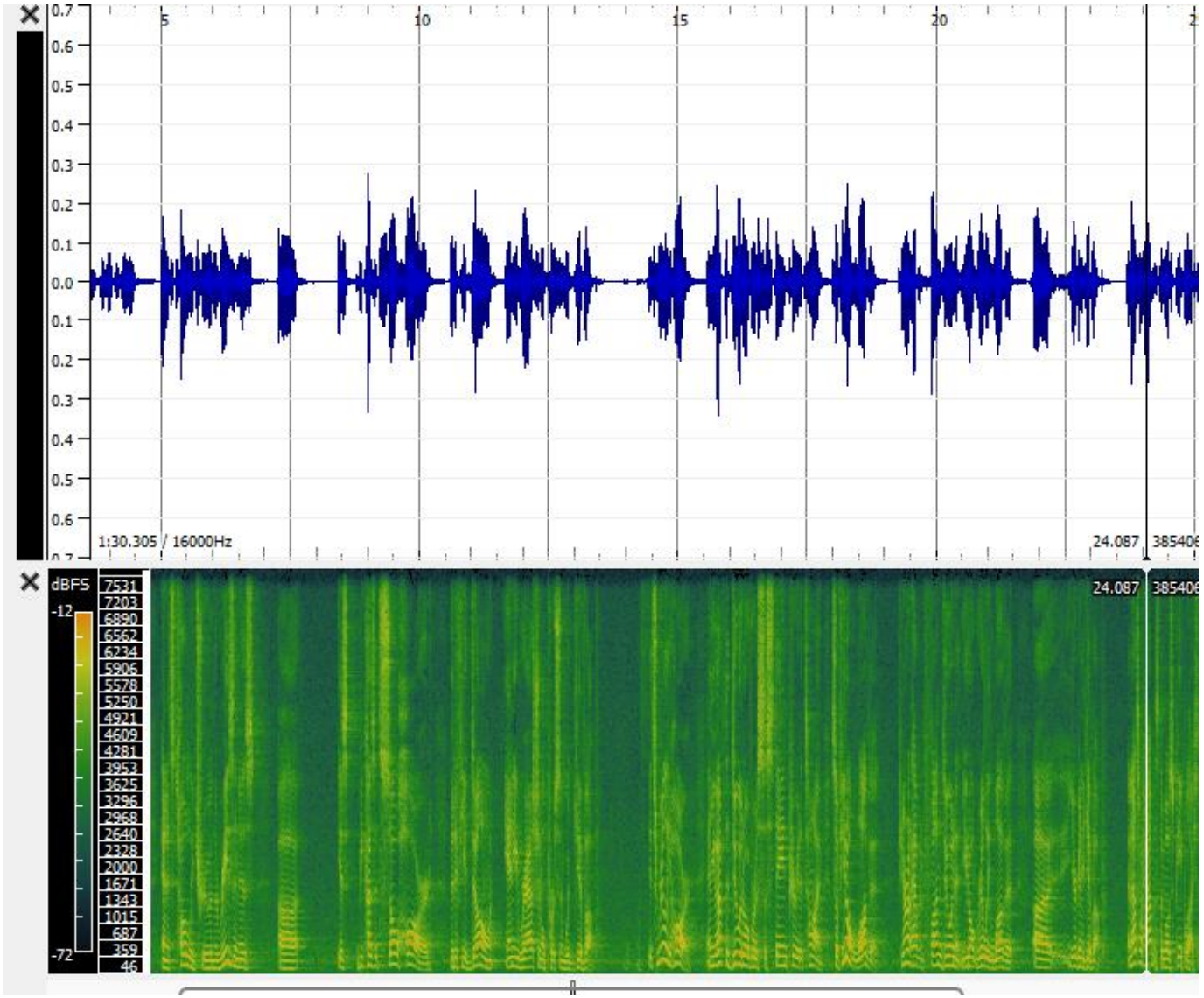
- Perform Fourier transform for each of the windowed segments
- Simple example first:

$$x(t) = \begin{cases} \cos(2\pi 10t) & 0\,\mathrm{s} \le t < 5\,\mathrm{s} \\ \cos(2\pi 25t) & 5\,\mathrm{s} \le t < 10\,\mathrm{s} \\ \cos(2\pi 50t) & 10\,\mathrm{s} \le t < 15\,\mathrm{s} \\ \cos(2\pi 100t) & 15\,\mathrm{s} \le t < 20\,\mathrm{s} \end{cases}$$



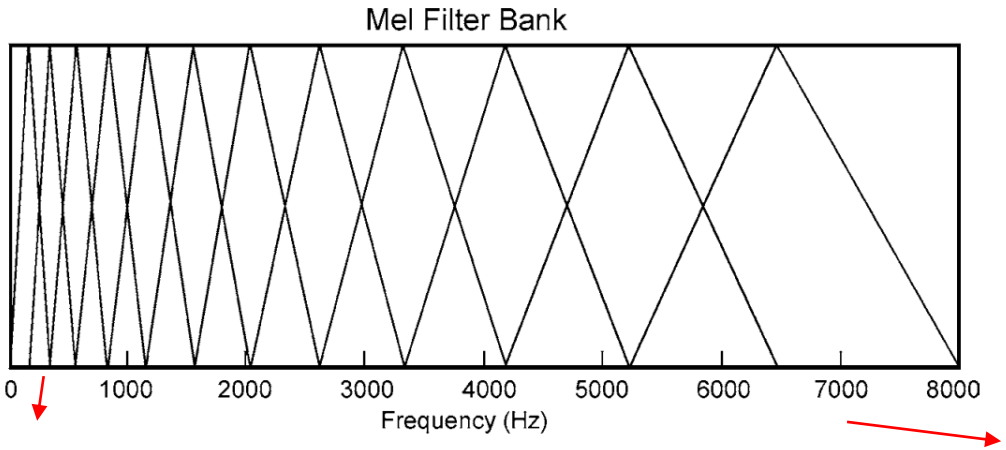
- This time-frequency plot is called spectrogram

Spectrogram



Mel filterbank

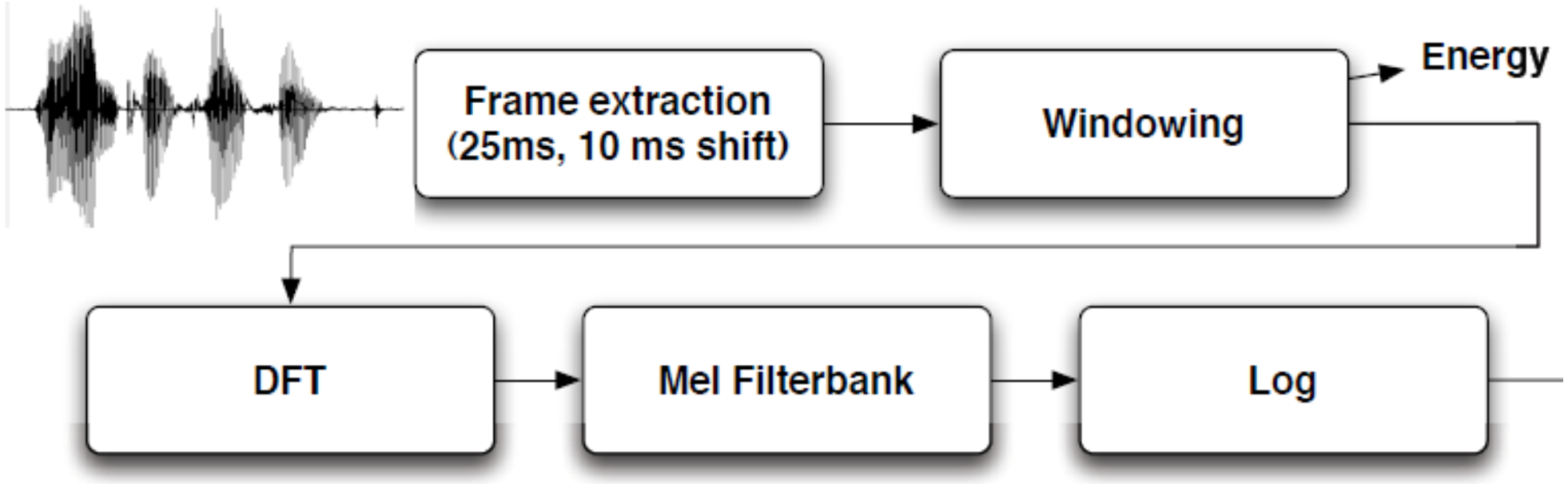
- Imitates human perception of speech
- State-of-the-art preprocessing step for applications like ASR, speaker recognition etc.
- Humans cannot properly distinguish two closely spaced frequencies
- This becomes more pronounced as the frequencies increase



More concerned about variations

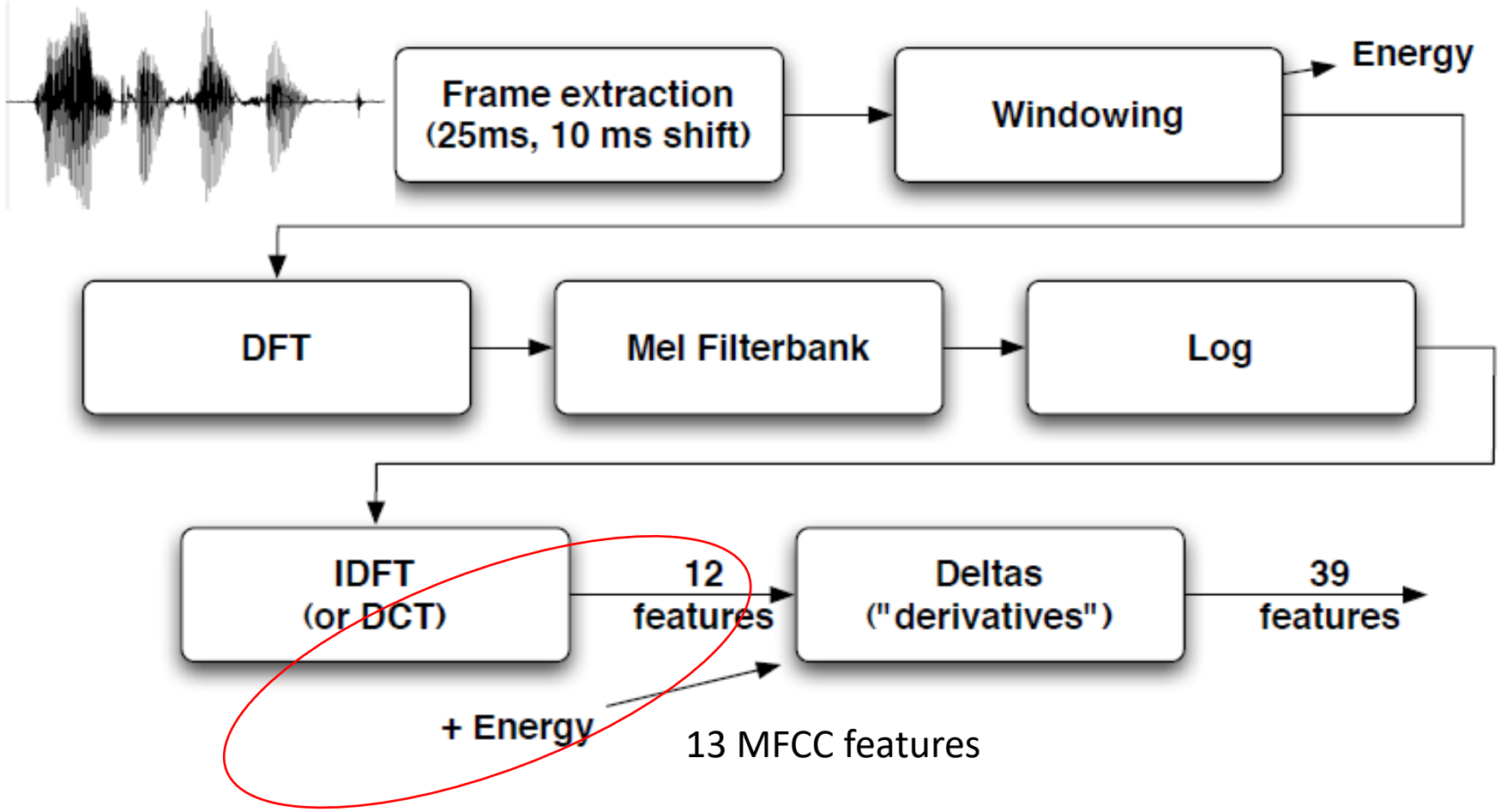
Less concerned about variations

Log scale



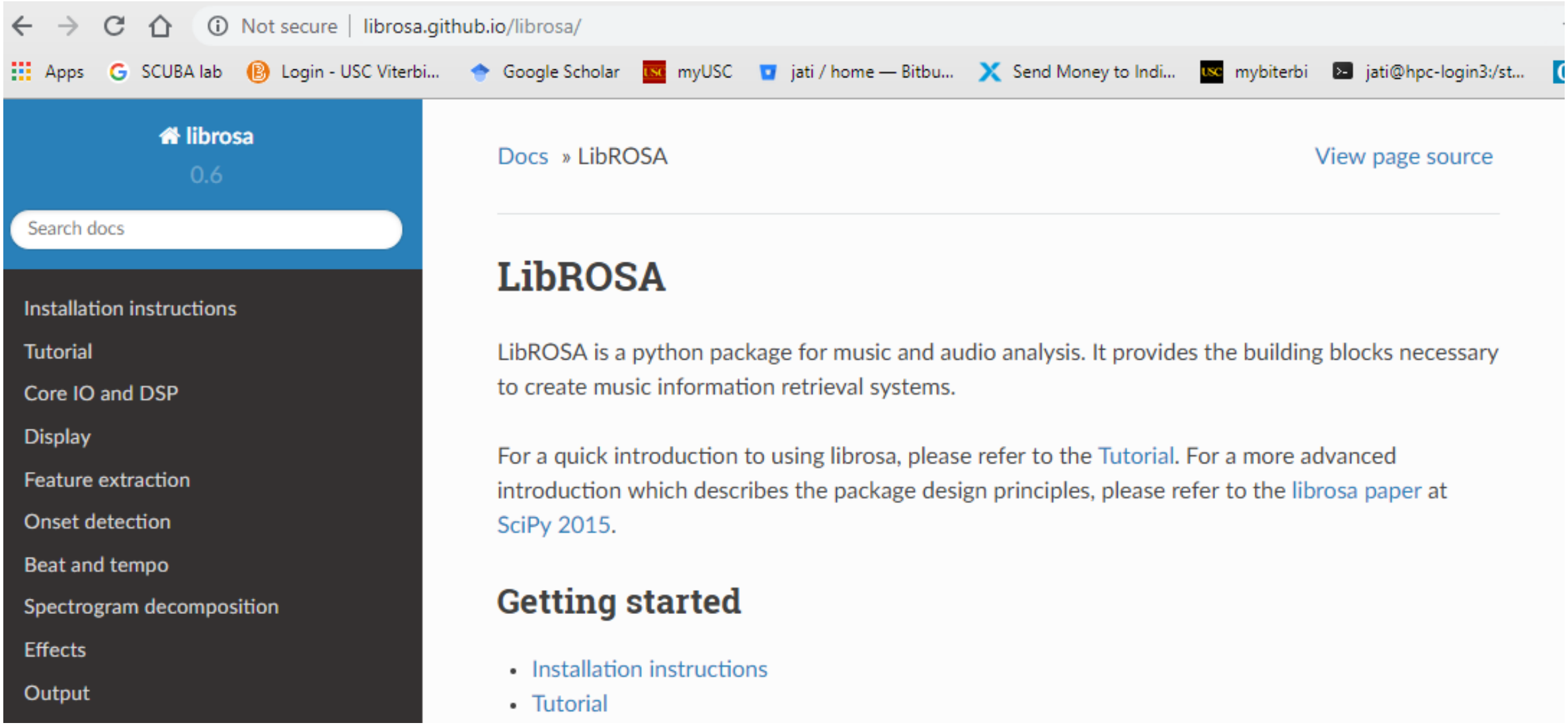
- Take log of the filterbank energies
- Motivated from human hearing perception
- We don't hear loudness on a linear scale
- Generally to double the perceived volume of a sound we need to put 8 times as much energy into it

Discrete Cosine Transform (DCT)



- DCT decorrelates the features
- Time domain derivatives are often employed as additional features

LibROSA


Not secure | librosa.github.io/librosa/
Apps
SCUBA lab
Login - USC Viterbi...
Google Scholar
myUSC
jati / home — Bitbu...
Send Money to Indi...
mybiterbi
jati@hpc-login3:/st...
librosa
0.6
Search docs
Installation instructions
Tutorial
Core IO and DSP
Display
Feature extraction
Onset detection
Beat and tempo
Spectrogram decomposition
Effects
Output
Docs » LibROSA
View page source
LibROSA
LibROSA is a python package for music and audio analysis. It provides the building blocks necessary to create music information retrieval systems.
For a quick introduction to using librosa, please refer to the Tutorial. For a more advanced introduction which describes the package design principles, please refer to the librosa paper at SciPy 2015.
Getting started
Installation instructions
Tutorial

Using LibROSA is easy!

You can utilize **Librosa** (`https://librosa.github.io/librosa/index.html`) to extract 64 dimensional MFCC features for all utterances. A sample code snippet is provided below:

```
import librosa
y, sr = librosa.load('audio.wav',sr=16000)
#sr should return 16000, y returns the samples
mat = librosa.feature.mfcc(y=y, sr=sr, n_mfcc=64, n_fft=int(sr*0.025), hop_length=
    int(sr*0.010))
print(y.shape, sr, mat.shape)
```

- This is configured for 25 m-sec frames and 10 m-sec skip.

LibROSA: Useful functions

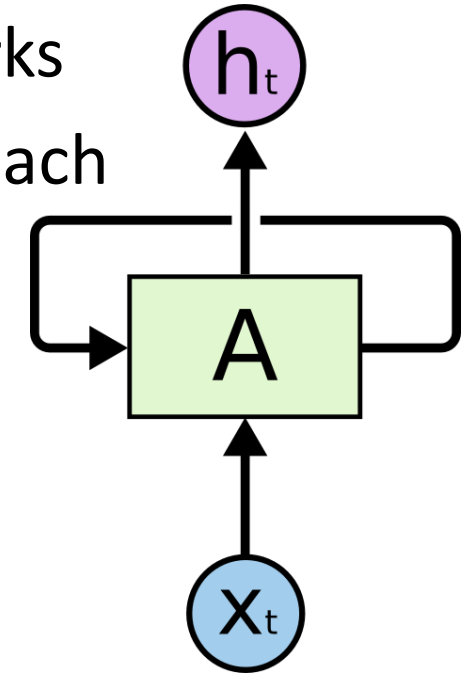
- librosa.core.load
- librosa.core.to_mono
- librosa.core.stft
- librosa.feature.melspectrogram
- librosa.feature.mfcc

- ➢Load audio
- ➢Converts to single channel audio
- ➢STFT
- ➢Gives mel filterbank features
- ➢Gives MFCC features

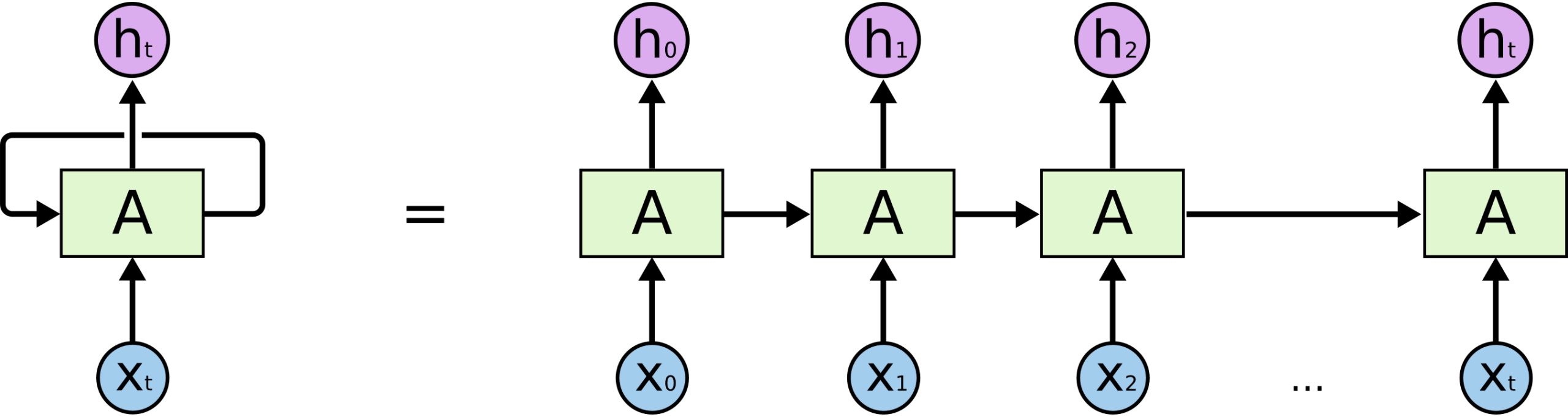
and many more …

Recurrent Neural Networks (RNN)

- Networks which contain traditional Neural Networks with loops in them so that information can persist
- These aren't all that different from traditional neural networks
- Can be thought of as multiple copies of the same network, each passing a message to a successor.



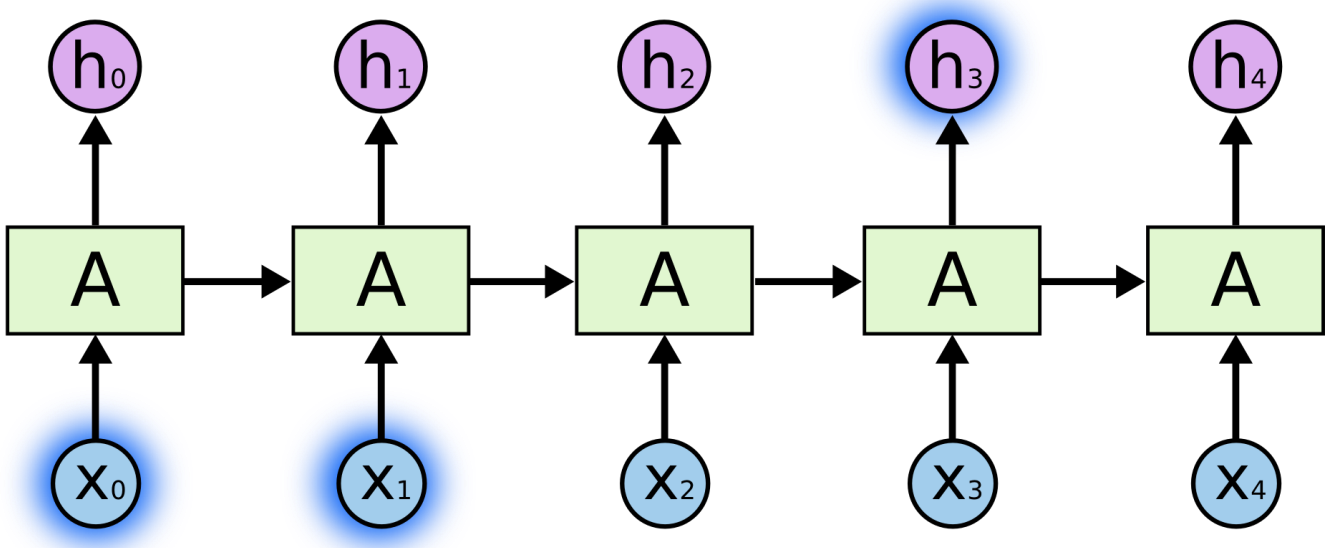
RNN Loop Un-rolling



Chain-like nature reveals that recurrent neural networks are intimately related to sequences

LSTMs – Special RNNs

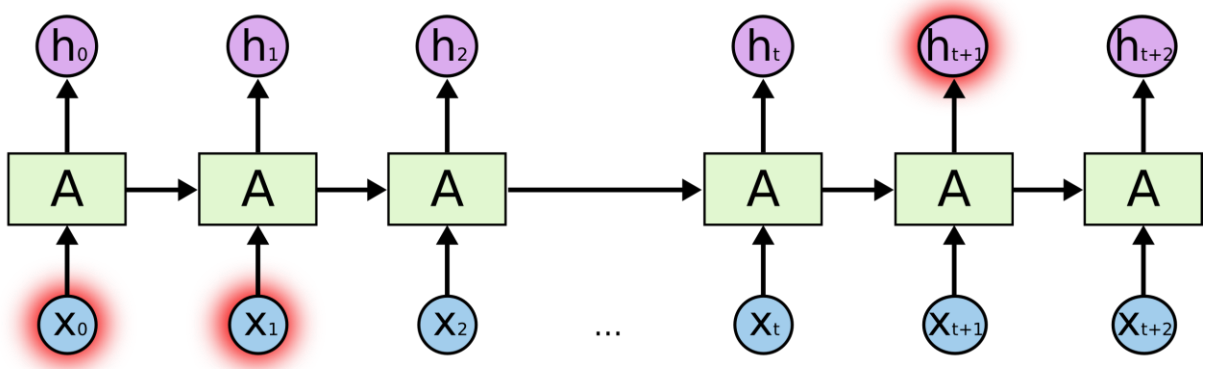
- Normal RNNs work well where the gap between the relevant information and the place that it's needed is small.



- Example: Language model trying to predict the next word based on the previous ones – "The clouds are in the *sky*"

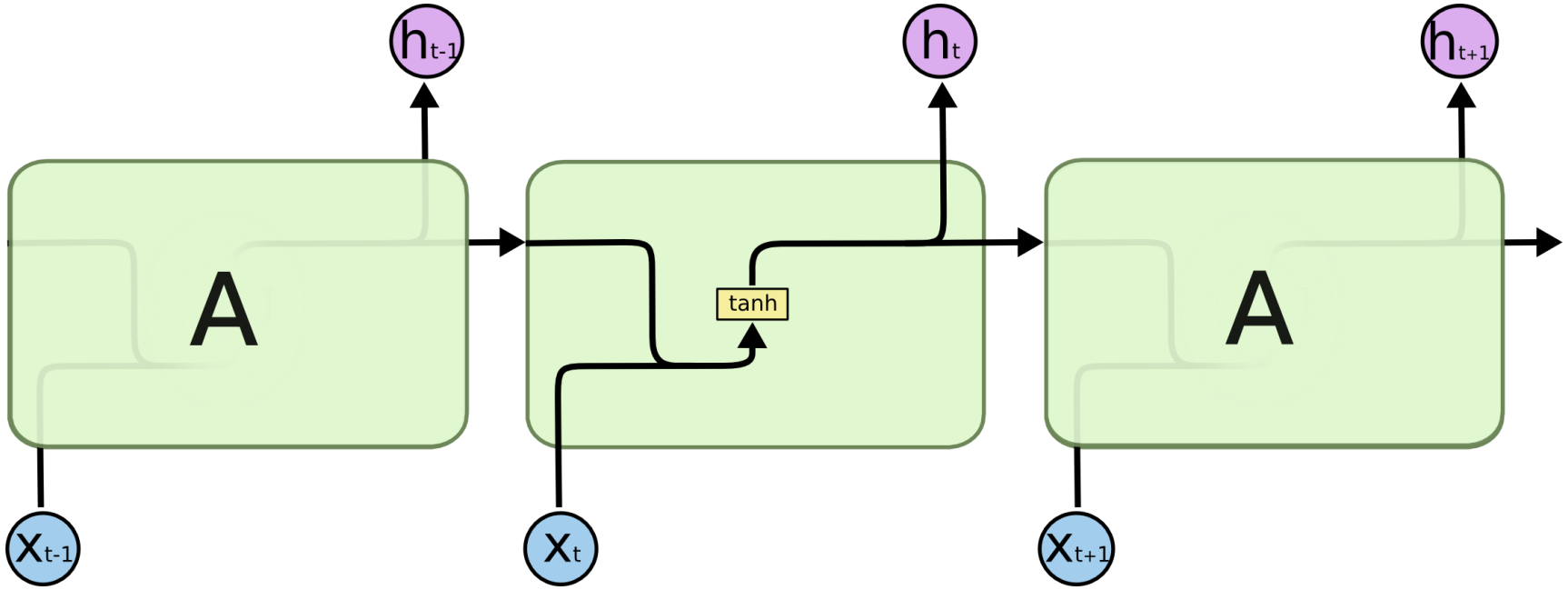
LSTMs – Special RNNs

- Sometimes more context is needed.
- Example: “I grew up in France… I speak fluent *French*.”
- Recent information suggests that the next word is probably the name of a language, but if we want to narrow down which language, we need the context of France, from further back.
- Entirely possible that the gap between the relevant information and the point where it is needed is very large.



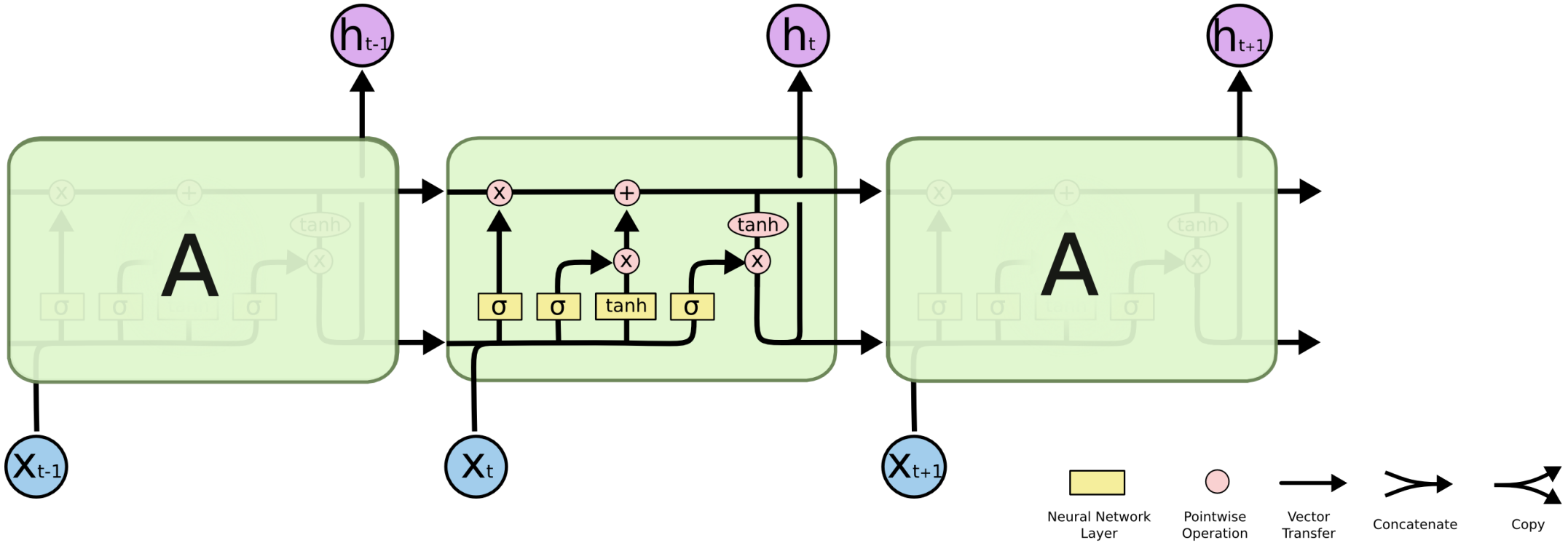
LSTMs – Special RNNs

- All recurrent neural networks have the form of a chain of repeating modules of neural network.
- In standard RNNs, this repeating module will have a very simple structure, such as a single tanh layer.



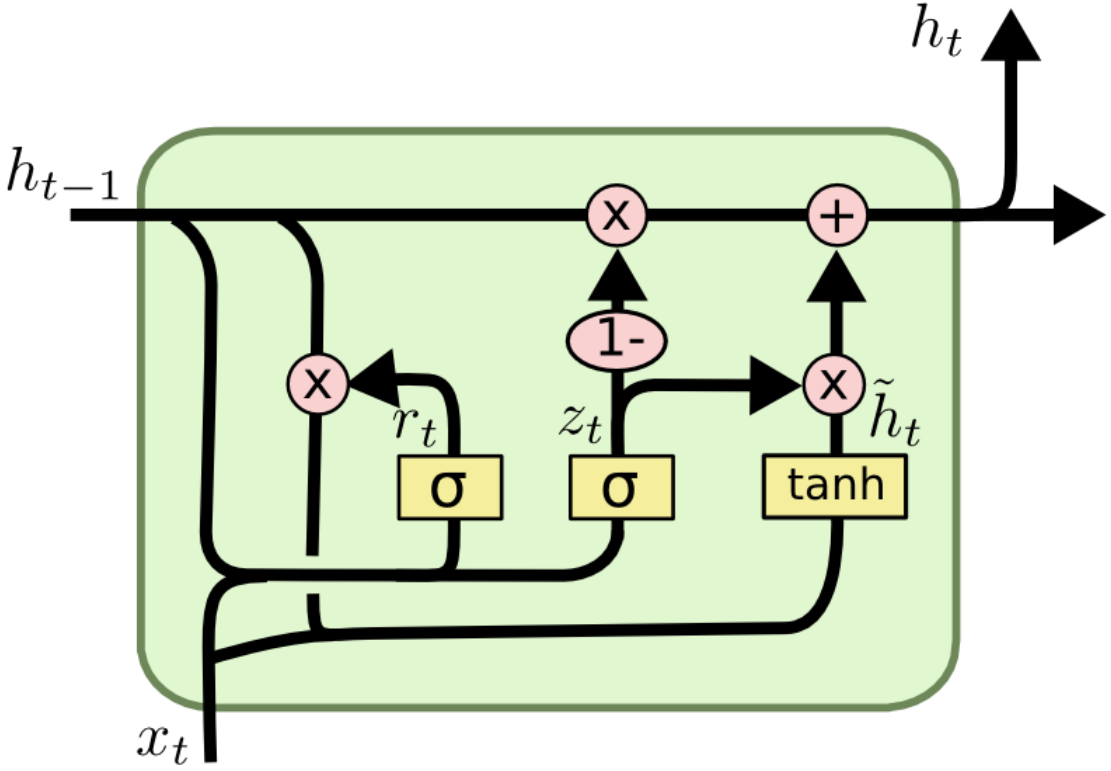
LSTMs – Special RNNs

- LSTMs also have this chain like structure, but the repeating module has a different structure.
- Instead of having a single neural network layer, there are four, interacting in a very special way.



Variation on LSTM – Gated Recurrent Unit (GRU)

- Simpler model than standard LSTMs
- Resources on LSTMs and GRUs (https://colah.github.io/posts/2015-08-Understanding-LSTMs/)



$$z_t = \sigma\left(W_z \cdot [h_{t-1}, x_t]\right)$$

$$r_t = \sigma\left(W_r \cdot [h_{t-1}, x_t]\right)$$

$$\tilde{h}_t = \tanh\left(W \cdot [r_t * h_{t-1}, x_t]\right)$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

Homework – 5

- Available on Piazza (cid @341)
- Tasks –
  - Extract Mel-frequency cepstral coefficients (MFCCs) from audio, which will be employed as features.
  - Implement a GRU/LSTM model, and train it to classify the languages. Input to this model will be the MFCC codes.