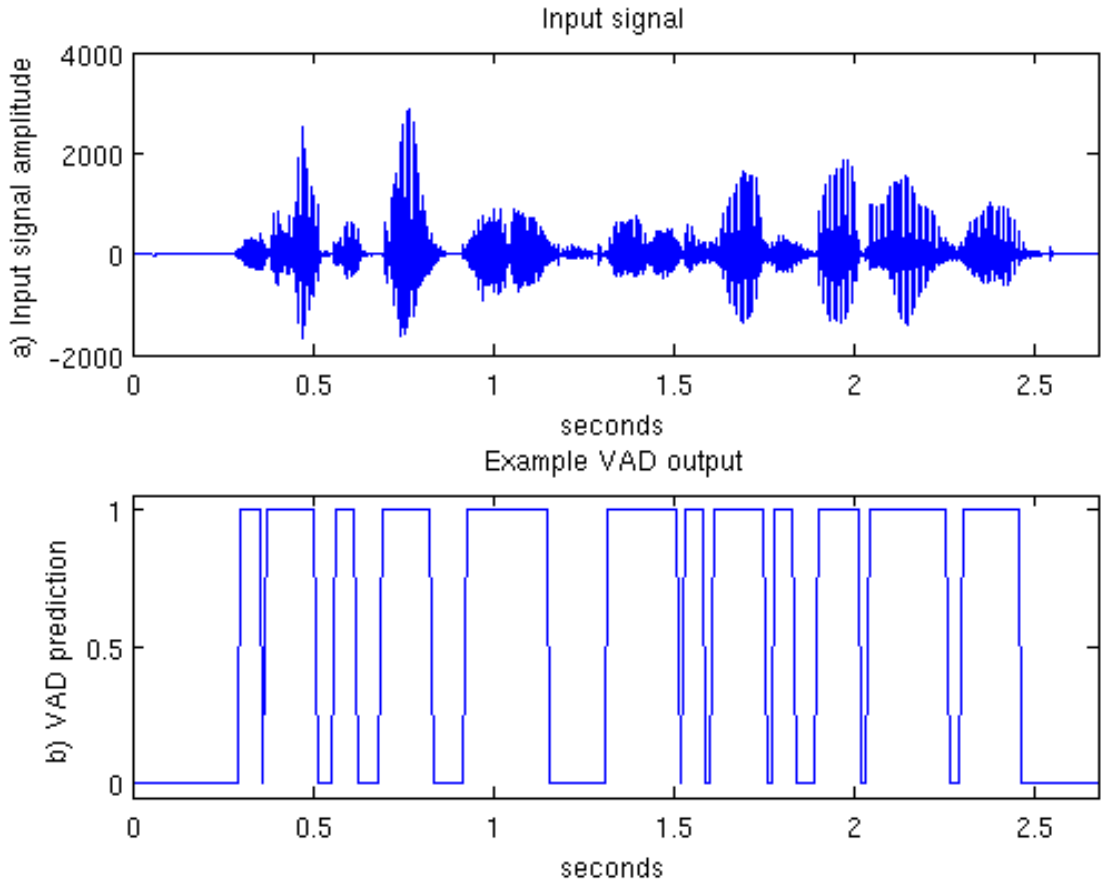
Voice Activity Detection (VAD) Tutorial

A Voice Activity Detector (VAD) is used to identify speech presence or speech absence in audio. This page will provide a tutorial on building a simple VAD which will output 1 if speech is detected and 0 otherwise.

The job of a VAD is to reliably determine if speech is present or not even in background noise. In perfectly clean conditions even a simple energy detector will do a perfect job at detecting speech; unfortunately you will never see perfectly clean signals in the wild. This means our VAD must be robust to noise.



Steps at a Glance

1. Break audio signal into frames.
2. Extract features from each frame.
3. Train a classifier on a known set of speech and silence frames.
4. Classify unseen frames as speech or silence.

Often a VAD will be used to classify voiced and unvoiced parts of speech as well as silence. The features introduced on this page are well suited to this task as well, but the classifier must have 3 classes instead of 2 (voiced speech, unvoiced speech and silence), otherwise everything else is the same.

Pre-Processing

The first step is to apply a high-pass filter to our audio signal with a cutoff around 200Hz. The purpose of this is to remove any D.C. offset and low frequency noise. There is usually very little speech information below 200 Hz so it does not hurt to suppress the low frequencies.

Prior to feature extraction our audio signal is split into overlapping frames 20-40ms in length. A frame is often extracted every 10ms. For example, if our audio signal is sampled at 16 kHz and we take our frame length to be 25ms, then each frame will have 0.025*16000 = 400 samples per frame. With 1 frame every 10ms, we'll have the first frame starting at sample 0, the second frame will start at sample 160 etc.

Feature Extraction

Once framed, we then extract features from each frame. We will follow Rabiner (2010, p595) in our choice of features. In the following discussion, $x(n)$ is a frame of speech, where n can range from 1 to L (the number of samples per frame). The following 5 features are extracted for each frame:

1. logarithm of frame energy:
$$E = \log\left(\sum_{n=1}^{L} x(n)^2\right)$$
2. zero crossing rate: the number of zero crossings per frame.
3. normalised autocorrelation coefficient at lag 1:
$$C = \frac{\sum_{n=1}^{L-1} x(n)x(n-1)}{\sqrt{\left(\sum_{n=1}^{L-1} x(n)^2\right)\left(\sum_{n=1}^{L-1} x(n-1)^2\right)}}$$
4. first linear prediction coefficient of a *p*th order predictor.
5. logarithm of linear prediction error from a *p*th order predictor

For this example we will use p=12 i.e. our linear predictor will be order 12.

Classifiers

No single one of our features will perfectly separate speech and silence frames, but by combining all the features in an intelligent way we can minimise the probability of error.

Rabiner uses a Bayesian classifier that consists of calculating the mean and variance of the features corresponding to silence and the mean and variance of the features corresponding to speech. To determine the label of an unseen frame, we calculate the likelyhood it comes from each label assuming the data is distributed according to the multivariate gaussian distribution. Whichever model gives the higher likelyhood is chosen as the frame label.

Alternatively we could use a discriminative classifier such as a Support Vector Machine (SVM). With a library like libsvm it is quite simple to train a SVM classifier for discriminating speech and silence.

Training

Training requires a set of files with known labels. This means a human has to go through the files and label each frame as speech or silence. This can be tedious, but the more training material the better the classifier will perform. An important detail is that the noise in the training files should match the noise in the testing files as closely as possible. If you cannot predict what noise will be present when using the VAD, then try and train with as many different noise types as possible at different SNRs.

If you want to apply a VAD to e.g. telephone speech, it is important all your training samples are recorded over the same channel as the one on which the VAD will be used. This will minimise any mismatch between training and testing. Once training is complete you should have a model that you can use to predict the labels of unseen features.

Putting it All Together

Once a model has been trained, we can determine whether speech is present or absent in new, unseen frames of audio. As the amount of noise increases, we expect the accuracy of our VAD to decrease.

Sometimes the predicted label can rapidly oscillate between speech present and speech absent. This behaviour is usually unwanted, and can be easily handled by applying a median filter to the predicted VAD labels.

If you have any questions or need anything clarified, don't hesitate to leave a comment.