Recognition of Parent Ragas with

Deep Learning

By Sripriya Barrow

Table of Contents

[1 Aim 4](#_Toc514018929)

[2 Background 4](#_Toc514018930)

[2.1 Motivation 4](#_Toc514018931)

[3 Literature Review 5](#_Toc514018932)

[3.1 AngularJS, A Birds eye view 5](#_Toc514018933)

[3.1.1 Why AngularJS 5](#_Toc514018934)

[3.1.2 A Glimpse into the Carnatic Realm of Ragas 6](#_Toc514018935)

[3.1.3 Neural networks and deep learning 8](#_Toc514018936)

[3.1.3.1 Neurons, the building blocks 9](#_Toc514018937)

[3.1.3.2 Weights and biases 9](#_Toc514018938)

[3.1.3.3 FeedForward & Back propagation Network 9](#_Toc514018939)

[3.1.3.4 Activation Function 10](#_Toc514018940)

[3.1.3.5 Data for Research 11](#_Toc514018941)

[3.1.3.5.1 How to get the Audio Data Required for the NN? 11](#_Toc514018942)

[3.1.3.6 Digital Representation of Sound 11](#_Toc514018943)

[3.1.3.6.1 Sound: 11](#_Toc514018944)

[3.1.3.6.2 How do we understand Sound? 11](#_Toc514018945)

[3.1.3.6.3 Sine Waves 14](#_Toc514018946)

[3.2 Artificial Intelligence and Machine Learning 15](#_Toc514018947)

[3.3 Cognitive Musicology 16](#_Toc514018948)

[3.4 Hacker’s guide to Neural Networks 16](#_Toc514018949)

[3.4.1 Real-values Circuits 16](#_Toc514018950)

[3.4.2 Circuits with Multiple Gates 18](#_Toc514018951)

[3.5 Shazam - a contemporary AI music tool 18](#_Toc514018952)

[3.5.1 Purpose: 18](#_Toc514018953)

[3.5.2 Analog to Digital conversion: 18](#_Toc514018954)

[3.5.3 How does Shazam work? 19](#_Toc514018955)

[3.5.3.1 The challenge 19](#_Toc514018956)

[3.5.3.2 How does Shazam achieve this? 19](#_Toc514018957)

[3.6 Deep Learning Models 20](#_Toc514018958)

[3.6.1 A Neural Network Model 21](#_Toc514018959)

[3.6.2 Input/Output formats 22](#_Toc514018960)

[3.6.3 Model Architechtures 22](#_Toc514018961)

[3.7 Other AI research projects 22](#_Toc514018962)

[4 Methodology 23](#_Toc514018963)

[4.1 Implementation Methodology 23](#_Toc514018964)

[4.2 Song Classification Methodology 26](#_Toc514018965)

[4.2.1 Training & Prediction: 26](#_Toc514018966)

[5 Objectives 27](#_Toc514018967)

[6 Resources 27](#_Toc514018968)

[7 Research ethics 27](#_Toc514018969)

[8 Activity Schedule 28](#_Toc514018970)

[9 References 28](#_Toc514018971)

# Aim/Abstract

The aim of this research proposal is to explore

1. the power of JavaScript focusing on AngularJS by implementing a system to
2. identify the parent Raga using Artificial Neural Network & Deep Learning.

Merge aims with objectives.

Research questions

# Background

## Motivation

Challenges I need to address.

[As a passionate developer, I love the scope of just developing something with technology. And JavaScript on one hand, from just being client side programming language and then conquering the server-side programming, it is a pure wonder that with so many evolving new technologies has just withstood the test of time and has amazed the development community with its sheer dynamics. AngularJS on the other hand is very popular among evolving JavaScript frameworks, but also is maintained by expert google engineers which makes this more tested and stable environment to work on, ofcourse, added with the simplicity it provides in applying JavaScript to Web applications with minimal cumbersome JavaScript code.]

Ragas are the basic scale used in Carnatic Music. Raga is the basis of a Classical Melody comprising of a combination of notes up and down the music octave (scale). Each raga has a unique scale ranging from 5 notes to 7 notes and the application of these notes to produce each melody brings the aspect of the raga in its rich form. Carnatic music is a type of Indian Classical Music originating from Southern India where my roots are firmly grounded.

There have been many researchers who have delved into this topic for over decades and there is so much more to learn and understand about the subtleties of the Music realm. Music and living being and more so, human beings, have a natural bonding and an intricate connection which is what attracts scientists and researchers alike to help understand this association. Many scientists intrigued by the connection between Music, a mathematical form that influence the emotional perception, an abstract form, have dedicated a lot of their time to research this concept.

On the other hand, Deep learning through Recurring Neural Network (RNNS) is a perspective of modelling real time objects taking inspiration from the human brain which is a wonder on its own.

Putting these pinnacles into one through this project will be a humbling effort and I believe a very deserving candidate for the master’s project.

# Literature Review

## AngularJS, A Birds eye view [background]

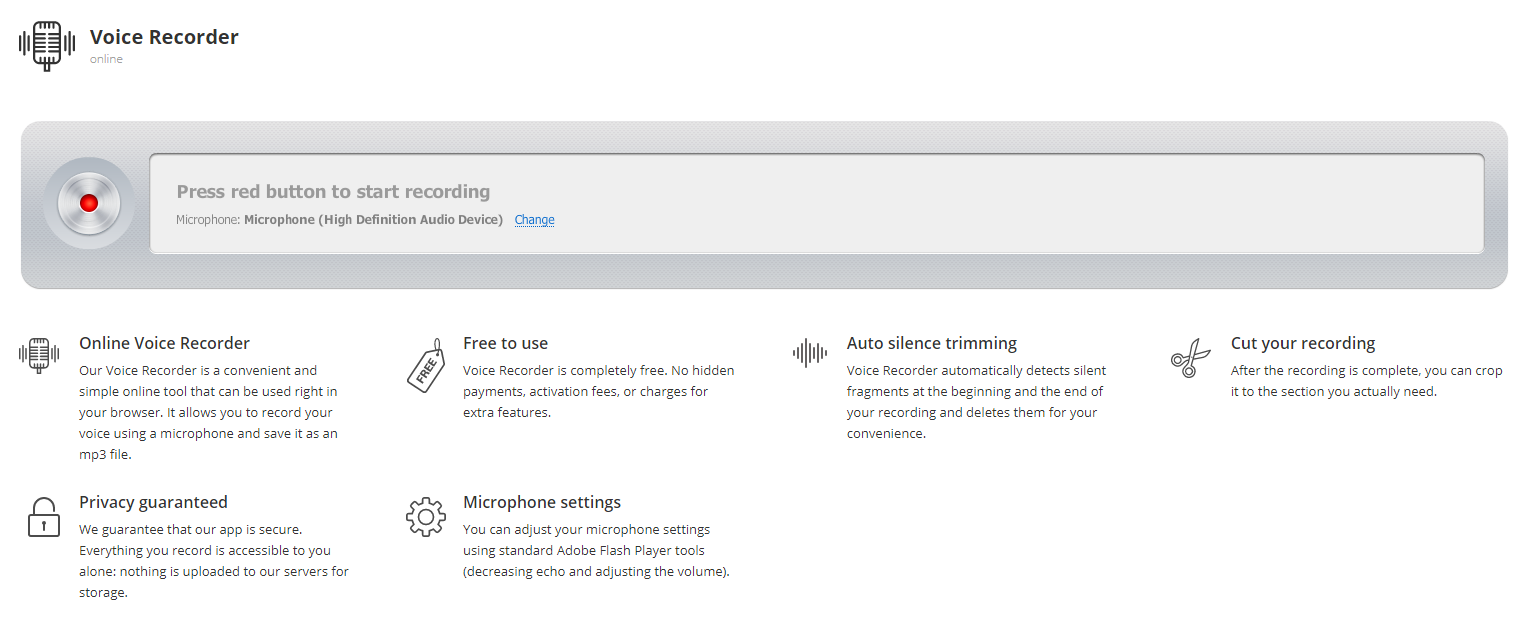
### Why AngularJS

In the Article published in “Journal of Global Research in Computer Science*” (Jain, Mangal, & Mehta, 2014)*, the following reasons are pointed why AngularJS is a favourable framework amongst most others:

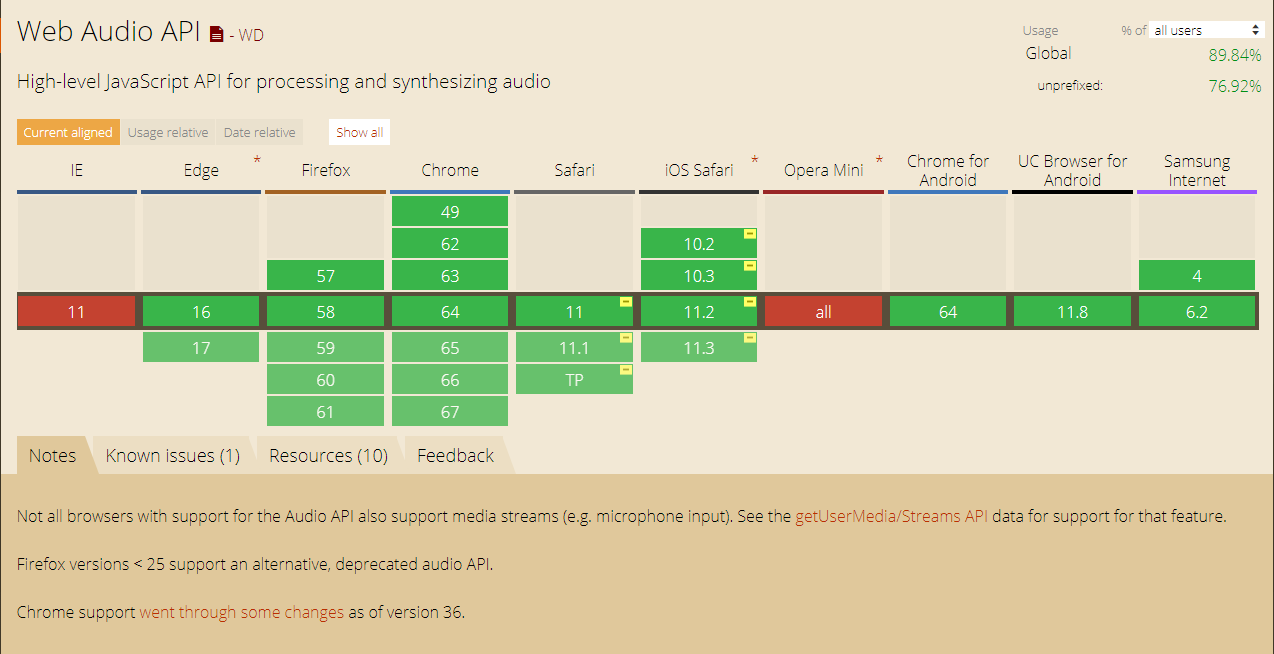
1. AngularJS is a JavaScript framework that extends HTML into concise, expressive and readable format
2. Handles dynamic content with simple markups, and leaves us to manage only the business logic.
3. With Single Page Application framework, Angular facilitates efficient code management.
4. Angular lives within the DOM environment and is POJO (Plain old JavaScript Objects)
5. Angular has backward compatibility and therefore works with browsers and older technologies, which is essential feature as Angular supports Old flash and HTML 5 format working with audio files.
6. Extensibility is key, as we will be looking at the Angular modules – angular recorder, wavesurfer.js and audiowaveform (c++) program to record music fragments for analysis though the Neural Network.

The front-end for this research will be a Web Application which allows users to record audio samples of music upto 30 - 40 second long and transforms this into a wave format. There will be a page for supervised training where the expected output will be input against each recordings. This will be the input for the neural network. The reason why we will implement the front-end web application in AngularJS is to forego the complexity of the HTML design and dynamic updates to concentrate on the logic behind the neural network and utilize the power of angular modules that enable audio recordings and mathematical extract for wave forms.

The audio recording web page will be a simplified version of the following web page (123apps LLC, 2018) but with visibile waveform and mathematical structure.



We will also be analysing the Web Audio API features, compatibility issues and the following chart, to understand and analyse the implementation of angularrecorder module.



### A Glimpse into the Carnatic Realm of Ragas

(Prasoon & Chakraborty, 2014). Indian Classical music is the art performed in the Indian subcontinent. The Origin of Indian classical music can be found in the Vedas, Samaveda in particular, which are the oldest scriptures in the Hindu tradition. The unique feature of the Indian classical music is, it is elaborate and expressive.

Very similar to the Western classical music, there are 12 semitones in an octave of which 8 basic are in the ascending order namely Sa, Ri, Ga, Ma, Pa, Da, Ni and Sa which in tonal effect match Do, Re, Mi, Fa, Sol, La, Ti & Do. Though Indian classical is equivalent in tonal quality to the Western classical system, Indian classical is monophonic while the western is polyphonic and therefore does not have the harmonies. Indian classical weighs on solo performance and puts weight on melody and rhythm and improvisation techniques. There are 2 forms of Indian classical – Hindustani and Carnatic. My research will solely be based on Carnatic, which is the art originating from Southern India.

There are 2 basic elements – Raga, the melodic structure and the tala, the rhythmic cycle. The raga is a melodic structure of ascending and descending notes with minimum of 5 notes in either order. The melakartha ragas are 72 permutation and combination with all 7 notes in both order (Santhanam, 2000). These 7 notes can be whole notes (whole numbers) or semi-notes (represented in fractions).

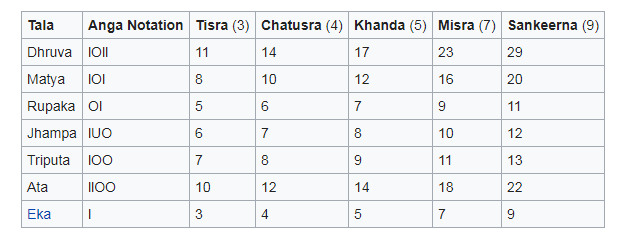
The concept of Raga is more complex that just notes. Raga requires certain set of notes with stress on the oscillations of specific notes which should invoke a certain emotional response altogether on use.

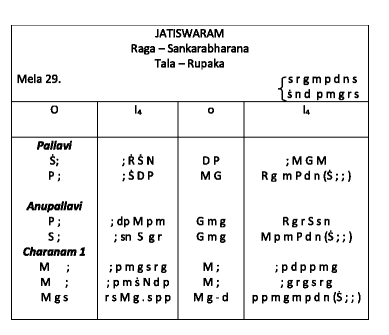
In western music, each measure has equal number of beats while in Indian Carnatic music each sub-division can have different number of beats constituting to a rhythmic pattern with a set number of beats in total.

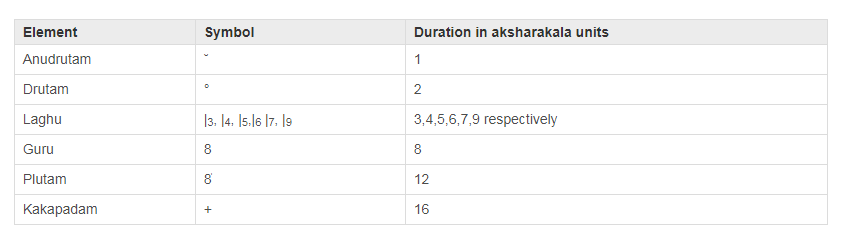
The western rhythmic pattern for a 4/4-time measure can be represented like this (Hein, 2014):



The above never changes during the rendition.

While in Carnatic music, talas are classified as below:

The measure of Rupaka Tala look like below (Mammen, Krishnamurthi, Varma, & Sujatha, 2016):

With subdivisions of the tala represented as below:

Carnatic music is close to nature as each raga is associated with time space such as morning, noon and evening ragas, dawn and dusk ragas, ragas before and after food, ragas pertaining to different seasons, ragas associated with emotions such as lullabies and mourning etc and much more.

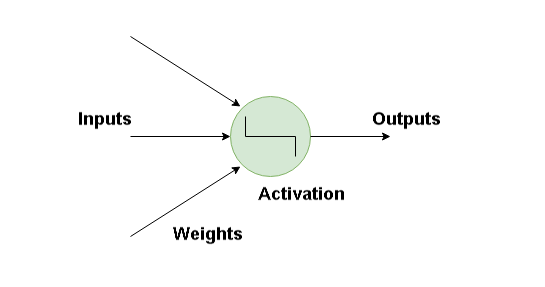
My research will include analysing recordings of each raga, deriving the notes within the recorded piece, and relating it to the training input for the neural network, to identify the closest Melakartha Raga for the recorded piece.

### Neural networks and deep learning

Deep Learning on Raga recognition has been researched previously though there is not too much information available in the public domain. Also, there are more research done using Python. Javascript based study is available more as cowboy developer trials. But, there are new tools that are written by many companies including Google’s deeplearn.js to apply machine learning and Artificial neural network in to practical use.

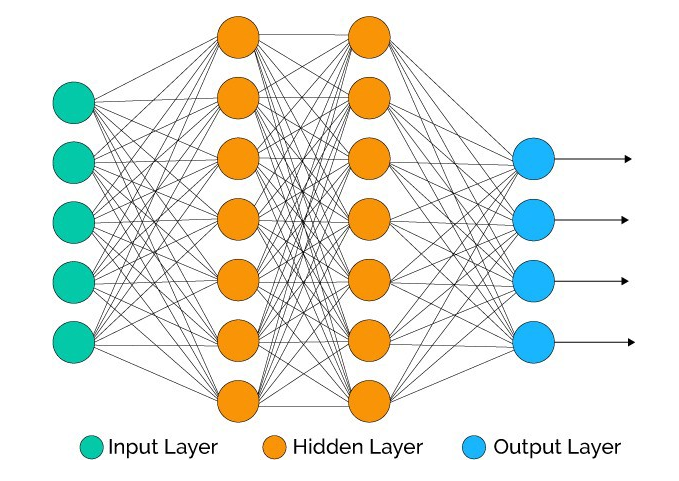
The reason behind deep learning it to be inspired by the human brain and its extraordinary power to make decisions. Thus, deep learning algorithm resemble the brain, as both involve a vast number of computational structures (neurons) which work in isolation to process a small information which is not significant in itself but produce extraordinary output when interacting with each other.

#### Neurons, the building blocks



The artificial neurons are the basic building blocks which in isolation processes a small proportion of information. The get inputs and bias (weights) which then produces an output signal using an activation function. There are multiple layers called the hidden layers between the input and the output layer within the neural network. (Kurama, 2018)

Deep learning contains Artificial neural networks with multiple layers including 1 input layer, 1 output layer and 1 or more hidden layers. As data travels through these layers it is processed and filtered to produce an output which becomes an input to another layer.

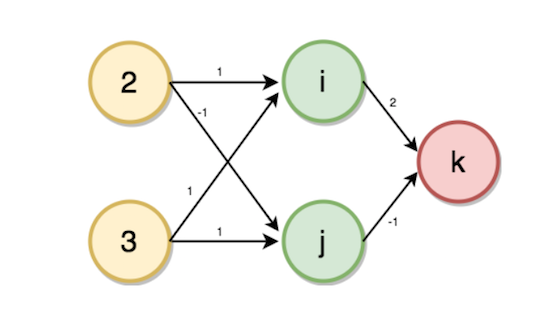


#### Weights and biases

Weights add strength to an input and resembles Linear regression where you give weights as coefficients. Weights are initialized to small random values from 0 to 1.

#### FeedForward & Back propagation Network

Feedforward supervised Neural network was the first successful learning algorithm in deep learning system. They are also known as multi-layer perceptron(MLP). Each neuron is connected to another neuron in the next layer with some bias. As the input is fed into the network, it moves forward activating each connected neuron to finally produce an output. This gets the name feed forward as the data flows in the forward direction. (Kurama, 2018)



In the above diagram, value of i and j will be calculated using the input values and weights from nodes 2 and 3 respectively.

i = (2 \* 1) + (3 \* 1)

→ i = 5

j = (2 \* -1) + (3 \* 1)

→ j = 1

k = (5 \* 2) + (1 \* -1)

→ k = 9

Back propagation algorithm is a method where the predicted value is compared againt the expected output to determine the error, which is then propagated back by updating the weights to reduce the error factor. This is propagated backwards from the output layer towards the input layer until the desired output is achieved.

#### Activation Function

To provide maximum prediction power to the neural network, we need to apply an activation function to the hidden layers. There are many different activation functions such as ELU, ReLU, LeakyReLU, Sigmoid, Tanh and Softmax. Understanding these activation functions and determining which one will suit the purpose of this research will be a planned task for further research.

As part of this research, we will also explore convolutional Neural Networks and Recurrent Neural Networks to determine the suitable method for achieving the outcome of this research and project.

#### Data for Research

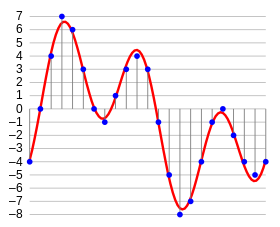
Faizan Shaikh talks about the complexity of handling unstructured data like audio or imagery. (SHAIKH, 2017) He states that unstructured data represents information that is closely related to human communication and interaction, which is useful and more powerful by content.

##### How to get the Audio Data Required for the NN?

The following formats are computer readable:

1. Wav (Waveform Audio File) format
2. Mp3 (MPEG-1 Audio Layer 3) format
3. WMA (Windows Media Audio) format

The visual representation of the audio format will look like this (SHAIKH, 2017):



#### Digital Representation of Sound

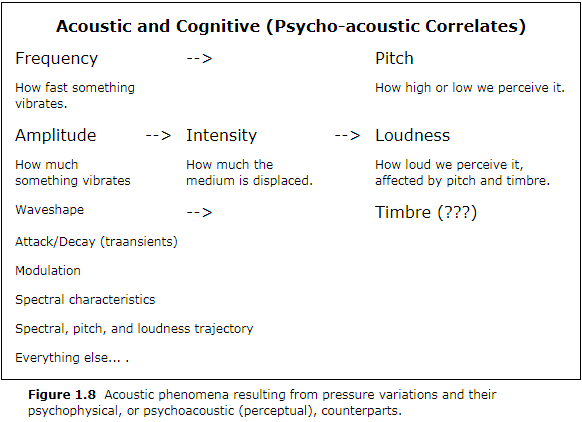
(Burk, Polansky, Repetto, Roberts, & Rockmore, 2018)

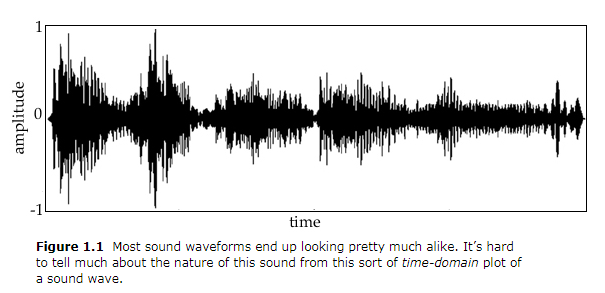
##### Sound:

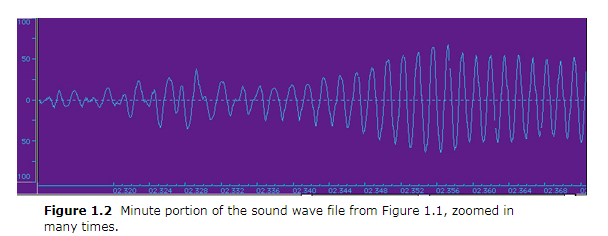
A sound is made up of vibrations that moves, that transmits causing generally air pressure variations which can be human audible within a given range. When we hear a sound, this is a result of sequence of events that had occurred in our human brain initiated by our eardrum.

##### How do we understand Sound?

To manipulate sound waves in the Computer, we need to understand Sound in many different ways. The following chart provides us with some insight of how we perceive terminologies relating to sound. (Burk, Polansky, Repetto, Roberts, & Rockmore, 2018)



To enable us to manipulate sound waves using Computer we need to understand the following:

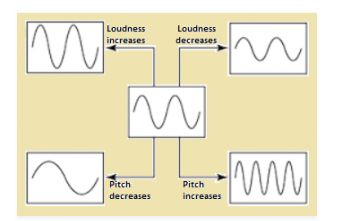


The above diagrams show the symmetry of sound waves which includes compression and rarefaction, or in other words what goes up comes down as defined in Newton’s third law of every action has an equal and opposite reaction.

The Y-axis represents amplitude which is the amount of air compression (above zero) and expansion or rarefaction (below 0). Zero is the rest position or pressure equilibrium (silence).

While Amplitude is mathematical, loudness is perception of sound. Frequency is the rate at which the air pressure fluctuates or repeats itself and the cognitive concept is pitch.

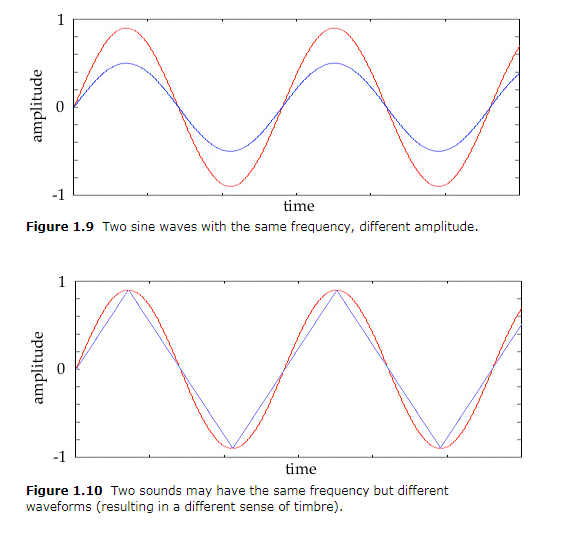
There is a clear and simple explanation of the difference between Pitch and loudness in byjus.com.



(BYJUS, 2018)

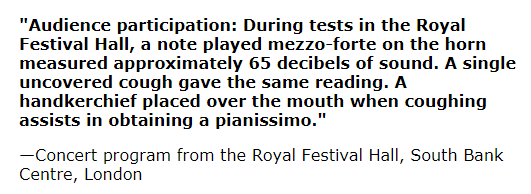
If you pluck a string in the guitar, and pluck it again harder, your pitch will the same, but your loudness will increase. This provides the relationship between pitch and loudness which are generally independent of each other.

The following diagrams (Burk, Polansky, Repetto, Roberts, & Rockmore, 2018) shows the difference in wave form:



Even though Amplitude is mathematical form of loudness, engineers use decibels to represent amplitude as human perceive loudness in decibels.

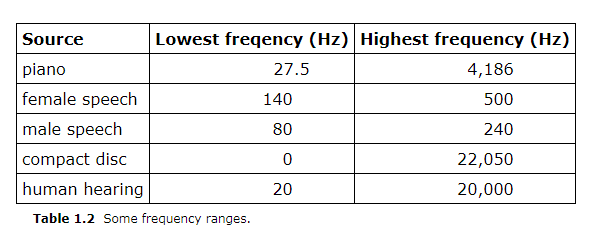
Decibels = 20 \* log(amplitude)



##### Sine Waves

A Sine wave is the simplest of waves and a very good candidate of repeating patterns of amplitudes, and the reason why it is called as simple harmonic motions. With an amplitude scale of 1 to -1, the sign wave goes from 0 to 1 to 0 to -1 and if this complete cycle of sine wave curve takes a second the frequency is one cycle per second or 1Hz (Hertz). (Burk, Polansky, Repetto, Roberts, & Rockmore, 2018)

The frequency range of sound of human hearing falls between 0 and 20kHz (kilo hertz or 20000 Hz). 0 – 20 Hz referred to as infrasonic range sound rather like beats as very low frequencies do not sound as a pitch.



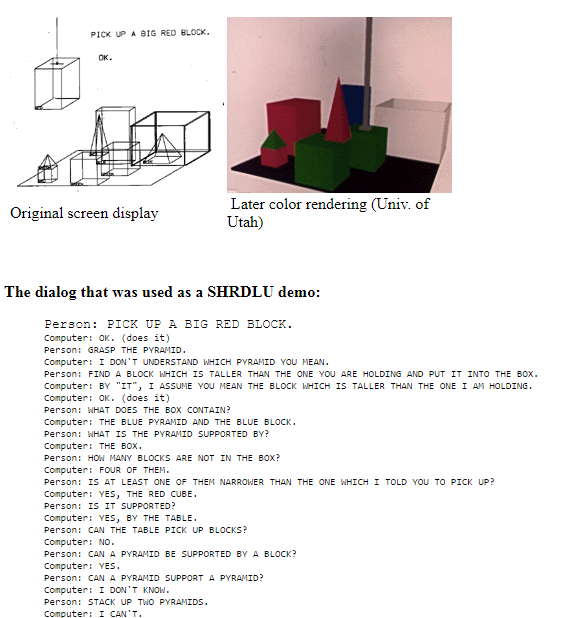
In music, frequency is referred to how often sonic event happens over time. A 20 Hz tone is a cycle repeating 20 times a second, which would mean a cycle, or a complete sine wave goes by in 1/20 of a second which is 0.05 second.

## Artificial Intelligence and Machine Learning

Artificial Intelligence and Machine Learning dates to 1950 when A.M Turing first questioned if “Computers can think?”. The seed for Artificial Intelligence and machine Learning was sown.

A.M Turing in his publishing, A. M. Turing (1950) “Computing Machinery and Intelligence. Mind 49: 433-460.” (Turing, 1950), quoted, “We may hope that machines will eventually compete with men in all purely intellectual fields. But which are the best ones to start with? Even this is a difficult decision. Many people think that a very abstract activity, like the playing of chess, would be best. **It can also be maintained that it is best to provide the machine with the best sense organs that money can buy, and then teach it to understand and speak English.** This process could follow the normal teaching of a child. Things would be pointed out and named, etc. Again, I do not know what the right answer is, but I think both approaches should be tried. “

SHRDLU is a program developed between 1968- 70 to understand the natural language that Turing mentions above, written by Terry Winograd at the MIT Artificial Intelligence Laboratory which takes AI to another interesting level. This was demonstrated using typed commands from human to computer to move simple blocks within the computer screen and the computer responds and asks questions to the human if the command is not clear (as shown below). Winograd used this for his dissertation with the title “Procedures as a Representation for Data in a Computer Program for Understanding Natural Language”. It was published as a full issue of the journal Cognitive Psychology Vol. 3 No 1, 1972, and as a book, Understanding Natural Language (Academic Press, 1972).(Winograd, 1971)



As Andrej Karpathy mentions quite often, connecting images and natural language played a very important role in the rapid development of Artificial Intelligence. The second important thing, he says, that made AI so powerful is availability of data. The third important thing that makes AI and machine learning possible is the technological advance that has created sophisticated infrastructure to help render faster and more able calculations.

## Cognitive Musicology

This is a collection of excerpts from the book Understanding Music with AI (1992) M. Balaban has synthesized

## Hacker’s guide to Neural Networks

This article by A. Karpathy is very useful and interesting as he specialises in Javascript library for training Neural Networks. He had developed a JavaScript library for training Neural Networks. As everyone agrees, Javascript is the best medium if you want a web-based application where browser enables easy visualization.

### Real-values Circuits

Real-valued circuits are fundamentals to engineering. These are binary gates that work with real values instead of Boolean values {0,1}.

Let’s take a single simple circuit with one gate, with real values (x=2, y=3).

\*

x

y

The above circuit takes 2 real variables x and y and with multiply gate returns x times y (2 \* 3 = 6).

Math representation of this will be f(x,y) = xy.

**Solution #1 (Random Local Search):**

We have 2 inputs x=-2 and y=3. We want to slightly increase the value by tweaking it. How do come about this? When we have only 2 inputs this is easy. We can apply a tweak of .01 to x and y and get a .05 improvement. But this is not as simple when we are dealing with millions of such inputs.

**Solution #2 (Numerical Gradient):**

The other solution is to push the output towards a positive improvement. This would then act on the x and y towards that positive move. But, a much simpler way is to iterate the inputs by one step in a loop and nudge the input slightly towards increasing the output. The change in the output is the derivative.

Mathematically,

Where h is tweak amount. The algorithm will be,

X = -2, y =3

output = function multiply(x,y) = -6

H = .01

//compute x-derivative

X’h = x+h

X’Output = multiply(x’h,y) = -5.9997

x-derivative =( x’output – output)/h = 3

//compute y-derivative

y’h = y+h

y’output = multiply(x,y’h) = -6.0002

y-derivative =( y’output – output)/h = -2

In practise, the value of h should be infintitely small as it should tend to go zero. But the above demonstration gives a good approximation. A derivative is used in respect to one input and the gradient in respect to multiple inputs. So, to say a gradient is made up of derivatives of all inputs.

**Solution #3 (Analytic Gradient):**

We can avoid the 100s of steps above and the approximation by using calculus to derive a direct expression that will be as straightforward as the output calculation. The analytic derivative requires no tweaking. So, with reference to our previous derivative formula, we will plug in the expression, f(x,y) = xy:

This makes the equation very interesting as the derivative x is just equal to y. The coincidence of this formula is that, in the previous working we tweaked x to x+h and it returned x-derivative as 3 which was the same as y and vice-versa with x = y-derivative = -2.

### Circuits with Multiple Gates

q

x

+

f

y

\*

**z**

The above results in the equation f(x,y,z)=(x+y)z. But since the final output is derived from the multiply gate, we can ignore the x and y and represent the above by equation f(q,z) = qz from the previous formulation. Also, q-derivative will be equal to z and z-derivative will be equal to q.

## Shazam - a contemporary AI music tool

This is an overview of Shazam as published in (Jovanovic, 2018).

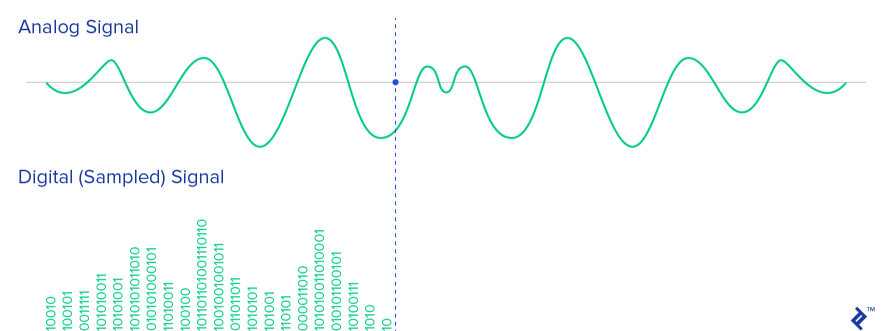
### Purpose:

Today, Shazam is the most popular and sophisticated AI tool that identifies a song. Now we all hear songs in the Radio or on the go, but cannot remember always what song it is? Shazam comes handy in identifying a song from a whole range of music collection within a music database.

There are many other programs and libraries in different programming languages written to achieve the Analog to Digital conversion of sound. Shazam is Java based and almost matches the objective of this dissertation.

### Analog to Digital conversion:

As input, Shazam converts the Analog signals to Digital. As Sound is the vibration created by air pressure and displacement through a medium, within the human ears, this vibration is transmitted to hair cells through the eardrum which then produces electrical impulses that are then fed to the brain through our auditory nervous system. Recording a song fairly produces a similar electrical signal. But, the sound wave is a continuous signal and the microphone also captures a recording as a continuous analog voltage signal, which is particularly not useful in the digital world. Therefore, before this can be processed by the computer, this needs to be translated into discrete signal which can be stored digitally. This is achieved by capturing the amplitude (the peaks or troughs at a given point in time).



As audible human frequency is 20 – 20KHz, the sampling Rate is always set to twice the frequency of 44100 Hz. Other settings that can be useful to analyse sound channels (1 for mono), sample size (16 bit).

The conversion of Analog-to-digital involves quantization of the input and therefore introduces a small error. The converters, to overcome this, perform multiple conversion on very small pieces of signal, a process called sampling.

### How does Shazam work?

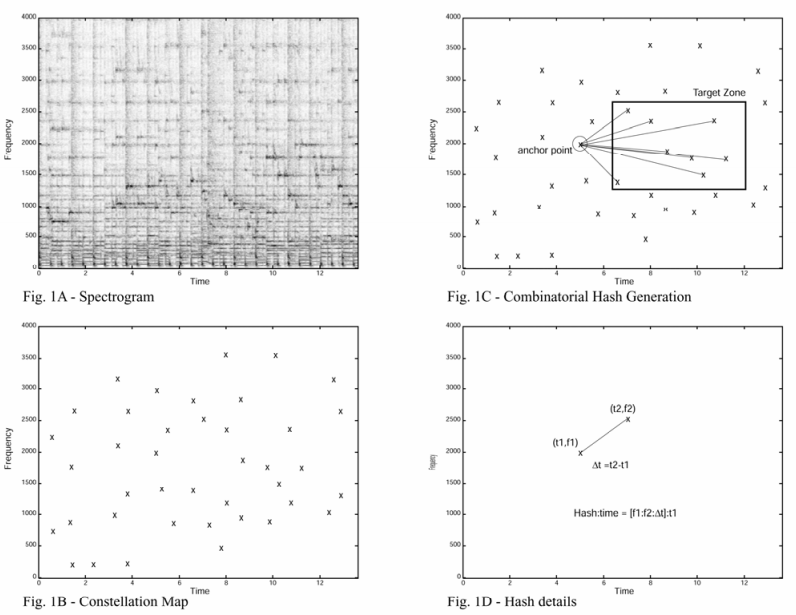
Sound Recording is easy in this age as there are so many Analog-to-digital converters. It is important to choose the correct API and programming language as Jovan Jovanovic puts it, that is important. In 2003, Shazam’s inventor Avery Li-Chung Wang published an article on Shazam algorithm (Wang, 2006).

#### The challenge

The algorithm had to recognise short audio samples captured with not so high end cellular microphone mixed with surround noise that is sent to the server across internet with limited bandwidth and subjected to Voice codec compression. After reaching the server, the algorithm is required to perform a search within a 2M song database and produce high accuracy and recognition rate.

#### How does Shazam achieve this?

Each audio file is fingerprinted. In this process, hash tokens are extracted which can be reproduced. Both sample files loaded by the user and the files in the database are analysed in the same way. The fingerprints are then matched for the samples against the database. Then the match is checked for correctness and accuracy.



## Deep Learning Models

Reference: https://www.youtube.com/watch?v=T0r-uCXvDzQ

Deep Learning is a young sub-field within Neural Networks which is a sub-field of Machine Learning. Neural network has only been in existence for 30 years and only in the last 8 years there have been good and viable design patterns constructed and they mostly are in the field of AI. Deep is the concept of the depth of the actual hidden layers that form part of the neural network.

Machine Learning

Neural Networks

Deep Learning

Some of the different models and purpose of deep learning used to resolve simple to complex problems are:

* Classification Models – Predict by identifying and classifying different features. Eg.: Google AutoML.
* Segmentation Models – Eg.: Auto driving where classification is done at pixel level.
* Image Captioning Models – This is to identify different objects in an image and describe the image.
* Generative Models – The neural network takes a blurred input image and produces a high resolution predictive image with more accuracy than a human brain. It can also produce coloured images from black & white images also known as the colourization model.
* Sequence Models – Text sequences and time series analysis are done using this model.

The above models are just a few. Other models include Speech Recognition, Translation, Anomaly detection, Q&A systems etc.

### A Neural Network Model

Input

Output

Hidden layer

Backpropogation/Optimizer

The building blocks above are:

* Inputs/Outputs
* Design patterns/algorithm
* Errors/Losses as the difference in expected output vs actual output
* Optimiser or weights/Biases

The optimizer can be built in to operate before the output or prediction, so the algorithm minimises the error in the output.

Data

Expected Result

Hidden Layer 1

Hidden Layer 2

Hidden Layer 3

Match

Expected

Result

Input

No

Yes

### Input/Output formats

* Arrays
* Vectorized Data
* Images – 3D images of scans, audio conversion to spectra
* Sequences – Text, Words, Speech, Images (videos, gifs), programs

### Model Architechtures

Model architectures

* *Fully Connected* – In this model each neuron is connected to every other neuron in the previous layer and each connection has its own weight. This is general purpose model and makes no assumptions on data features. This is the most expensive in use of memory (weights) and calculations (connections).
* *Convolutional* – Convolutional Layer is more advanced and provides specialization in handling data. Each neuron is connected only to a few other neurons which are located nearby. The most apt candidate for convolutional Model is Dense Captioning where you identify features in an image data. Same set of weights are shared by all neurons and can be equated to filters that identify specific features. More efficient in terms of memory and computation.
* *FeedForward* – Feedforward is a basic recurrent network also known as multilayer perceptron (MLP). A feedforward network defines y=f(x,), by mapping input x to category y and learning the value of parameter . The network is unidirectional and only flows from left to right.
* *Recurrent Neural Network -* When these network extend to include feedback connections they are called recurrent neural networks.

## Other AI research projects

# Methodology

For this research project, our training data-set will be the music notes or swaras that makeup the 72 melakartha ragas which will be created based on the table available in <http://www.carnatic.com/carnatic/ragalist.htm>. As these are parent Ragas and all other ragas will be a subset of these, the focus will be to model the ANN to infer the closest parent raga.

The basic mapping of notes will be denoted as represented in the below table:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| # | Name | Carnatic | Symbol | Solfege | Piano Key |
|  | Shadjam | Sa | S | Do | C |
|  | S Rishabh | Re | R1 | Re (flat) | D♭ |
|  | C Rishabh | Re | R2 | Re | D |
|  | S Gandhar | Ga | g1 | Mi (flat) | E♭ |
|  | Sa Gandhar | Ga | G2 | Mi | E |
|  | S Madhyam | Ma | M1 | Fa | F |
|  | P Madhyam | Ma | m2 | Fa (sharp) | F# |
|  | Pancham | Pa | P | Sol | G |
|  | S Daivat | Dha | d1 | La (flat) | A♭ |
|  | C Daivat | Dha | D2 | La | A |
|  | Kai Nishad | Ni | N1 | Si(flat) | B♭ |
|  | Ka Nishad | Ni | N2 | Si | B |

To achieve this, we will record samples of various music pieces into wav files and this will be the input to our ANN. The ANN then processes, identifies the notes in the piece and analyses the patterns to identify the closest Raga.

The process we will use to feed into our NN:

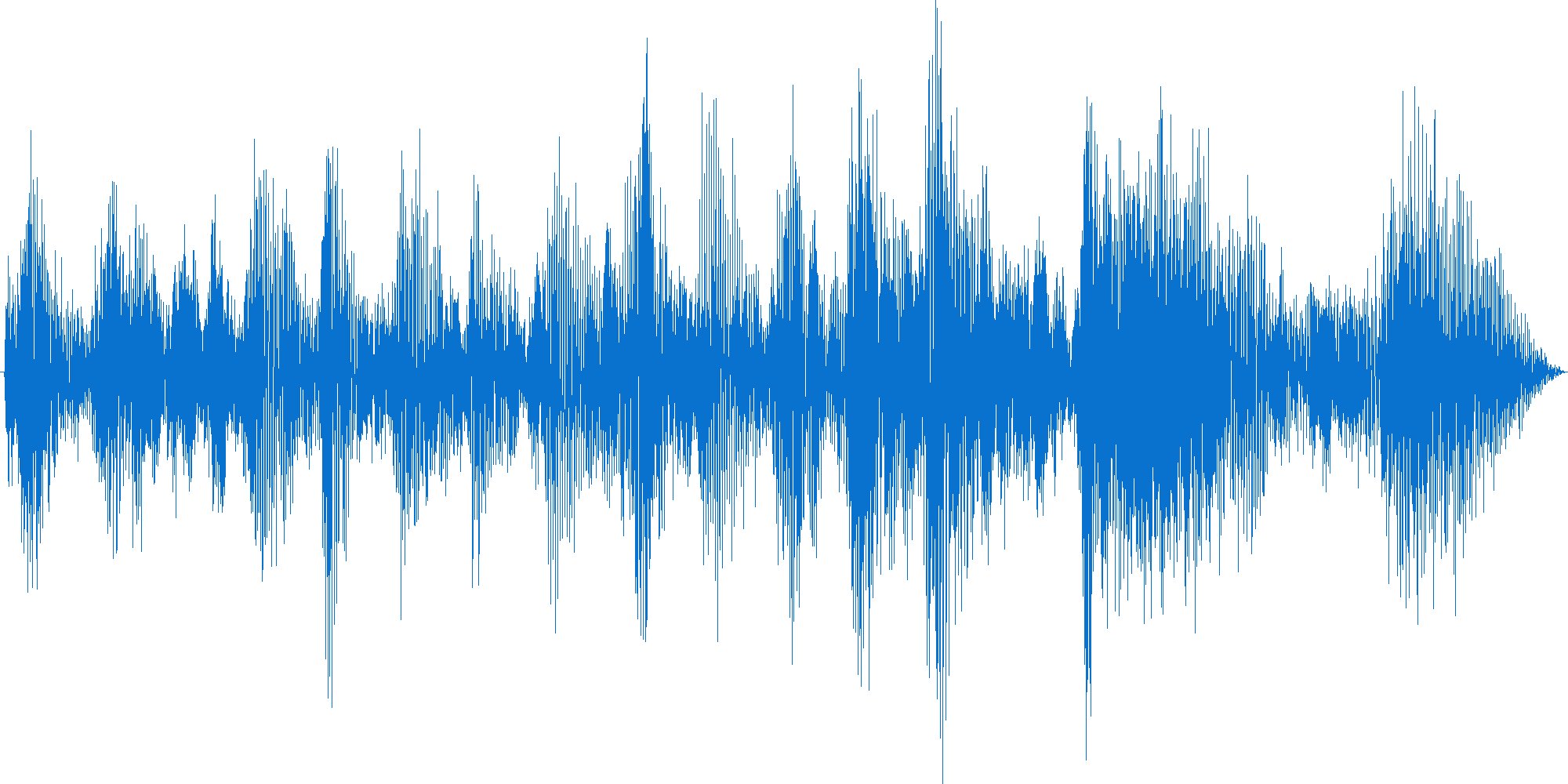
1. Angular Recorder to manually record Ragas sung by different musicians into wav form. The source will personal recordings from self-play and recordings from YouTube and other audio available in the public domain for free. We will collect 10 times 30s fragments of different singers singing 10 Ragas as sample.
2. We use wavesurfer.js to play these recordings.
3. We will use audiowaveform, which is a c++ program to then save the wave information into json format, which will be our input to our NN.

The music to notation translation model will be designed based on Dean Malone’s app (Malone, 2017) which is based on the Verovia Javascript toolkit. Each recording will be translated to notation and then compared with the Raga notations of the melakartha ragas.

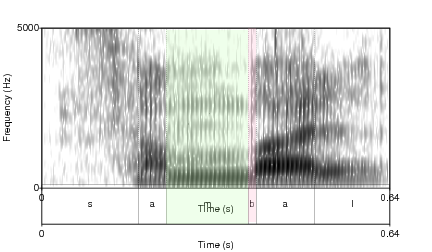
## Implementation Methodology

https://hackernoon.com/finding-the-genre-of-a-song-with-deep-learning-da8f59a61194

1. Audio to be recorded into wave form (40 second)



1. Wave is transformed to image (spectrogram)



1. Spectrogram is sliced further into 5s data. This gives us 20 sets of data from each recording. If we take 5 ragas to train and 10 recordings for each raga, we have 10 \* 40 = 400s of audio converted to 400 \* 20 = 8000 sets of data for our neural network.

Note: As there is more reliable codebase in image processing, this will enable us to reuse code base from other researchers, to avoid reinventing the wheel.

1. Out of the 8000, input data, 70% of this data will be used for training. (8000 \* .70 = 5600 sets, i.e., 560 sets per raga). 20% will be used for validation. (8000 \* .20 = 1600 sets, i.e., 160 per raga). 10% will be used for testing. (800 sets i.e. 80 per raga).
2. For recording and display we will use the WebAPI and Javascript (Angular) to enable users see the output of the recording in the browser. Verovio is a good candidate for conversion to notation, which will be used if we do display the audio recording by notes.
3. The spectrogram slices will be used as input to our neural network which will be developed using Tensorflow.js or Deeplearn.js.
4. The browser will then be used to display the predictions from the neural network (either Deep Convolutional Neural Network or Recurring Neural Network Algorithm will be used).

## Song Classification Methodology

### Training & Prediction:

Input

Hidden Mesh

Image Slices



Expected Result (train)

Read, Match, Infer

Data Store

Probability 1 – Raga1

Probability 2 – Raga5

Probability 2 – Raga1

Predictions will be indicative based on the voting results (best of 3). If voting results are ambiguous, the prediction will be “I am not sure”.

Prediction : Raga 1, for sure.

Voting

# Objectives

The purpose of this research is to explore deep learning using artificial neural networks and implement a web app using AngularJS which will allow users to record audio samples and to create input notations to be fed into the neural network for Raga prediction. To complete this research and development the following objectives will be met:

1. Critically evaluate existing work on machine learning and artificial neural network using.
2. Collect data by recording audio samples of Carnatic music rendering which will include self-rendition and sources available in the public domain for free, mainly, sourced from YouTube. Sampling will be conducted on 10 Ragas.
3. Develop UI to translate recordings to notation, which will be stored in the database using AngularJS.
4. Develop business logic using deeplearn.js to create predictive model using artificial neural network. Develop UI to feed input data through atleast 2 networks and to display prediction.
5. Critically evaluate the strategies and tools used for the proposed model against other options and assess pros and cons of the implemented model.
6. Draw conclusions and recommendations and produce dissertation thesis.

# Resources

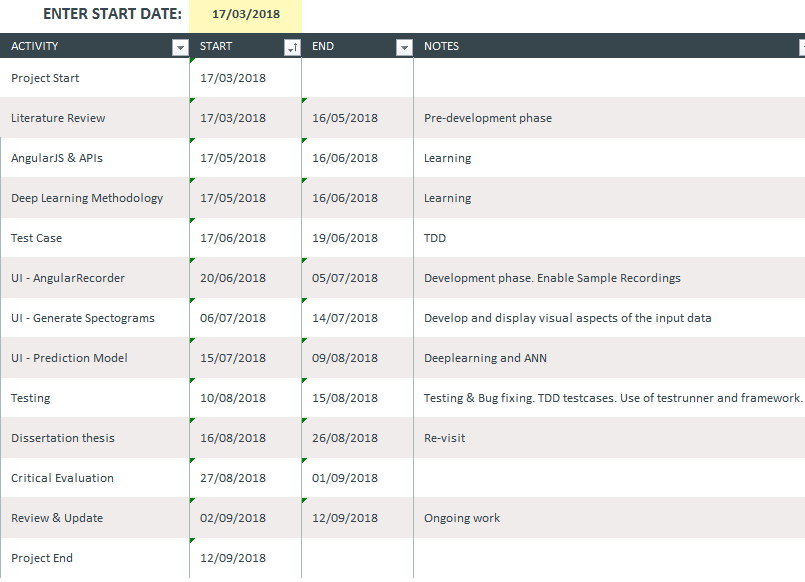
The data used for this research will be recordings of self-rendition and audio recording samples through the modelled web application sourced mainly from YouTube or another public domain available free for use.

The music notations will also be generated through the UI of the proposed model which will be used as input for the prediction model.

# Research ethics

There will be no private or personal information collected, stored or used for this research purpose. All sample recordings collected will be self-rendered or sourced from YouTube or those which are available in the public domain and free for use. All data sources will be clearly disclosed in the dissertation thesis. However, inadvertently, if any data collected is recognised to have confidential information, this will be excluded from the dataset.

# Activity Schedule



# References

123apps LLC. (2018). *https://online-voice-recorder.com/*. Retrieved from https://online-voice-recorder.com/: https://online-voice-recorder.com/

Burk, P., Polansky, L., Repetto, D., Roberts, M., & Rockmore, D. (2018, March 11). *Music And Computers*. Retrieved from http://sites.music.columbia.edu/: http://sites.music.columbia.edu/cmc/MusicAndComputers/

BYJUS. (2018, Mar 14). *Sound Waves - Loudness | Types of Pitch | Difference*. Retrieved from https://byjus.com/: https://byjus.com/physics/loudness-of-sound/

Hein, E. (2014, November 30). *Visualizing rhythm*. Retrieved from The Ethan Hein Blog: http://www.ethanhein.com/wp/my-nyu-masters-thesis/visualizing-rhythm/

Horwood, K. W. (2013). *Musical Keyboard - JS Dynamic Audio Synth*. Retrieved from http://www.keithwhor.com/: http://www.keithwhor.com/music/

Jain, N., Mangal, P., & Mehta, D. (2014). AngularJS: A Modern MVC Framework in JavaScript. *Journal of Global Research in Computer Science, Volume 5*, 13.

Jovanovic, J. (2018, 05 12). *shazam-it-music-processing-fingerprinting-and-recognition.* Retrieved from https://www.toptal.com/: https://www.toptal.com/algorithms/shazam-it-music-processing-fingerprinting-and-recognition

Kurama, V. (2018, March 14). *Deep Learning with Python*. Retrieved from Towards Data Science: https://towardsdatascience.com/deep-learning-with-python-703e26853820

Malone, D. (2017, January 29). *Piano Play: Angular 2 piano notation app*. Retrieved from dean Malone: http://deanmalone.net/post/piano-play/

Mammen, S., Krishnamurthi, I., Varma, A. J., & Sujatha, G. (2016, February 16). *EURASIP Journal on Audio, Speech, and Music Processing*. Retrieved from Springer Open: https://asmp-eurasipjournals.springeropen.com/articles/10.1186/s13636-016-0083-z

Prasoon, P., & Chakraborty, S. (2014). *Raga Analysis Using Artificial Neural Network.* Lambert Academic Publishing.

S. Samsekai Manjabhat, S. G. (2017). *Raga and Tonic Identification in Carnatic Music*. Retrieved from Journal of New Music Research: https://www.tandfonline.com/doi/full/10.1080/09298215.2017.1330351?src=recsys

Santhanam, K. (2000, October 15). *Carnatic - Raga List*. Retrieved from http://www.carnatic.com/: http://www.carnatic.com/carnatic/ragalist.htm

SHAIKH, F. (2017, August 24). *Getting Started with Audio Data Analysis using Deep Learning (with case study)*. Retrieved from https://www.analyticsvidhya.com/: https://www.analyticsvidhya.com/blog/2017/08/audio-voice-processing-deep-learning/

sk89q. (2018, March 14). *Dynamic .WAV Generation in JavaScript*. Retrieved from https://www.sk89q.com/: https://www.sk89q.com/playground/jswav/

Turing, A. M. (1950). *Turing.pdf.* Retrieved from https://www.csee.umbc.edu/: https://www.csee.umbc.edu/courses/471/papers/turing.pdf

Wang, A. L.-C. (2006, Jan 19). *An Industrial-Strength Audio Search Algorithm .* Retrieved from http://www.ee.columbia.edu: http://www.ee.columbia.edu/~dpwe/papers/Wang03-shazam.pdf

Winograd, T. (1971, 02). *SHRDLU.* Retrieved from https://hci.stanford.edu/: https://hci.stanford.edu/winograd/shrdlu/