Analyses of Convolution Neural Networks for automatic tagging of music tracks

Aravind Sankaran

RWTH Aachen

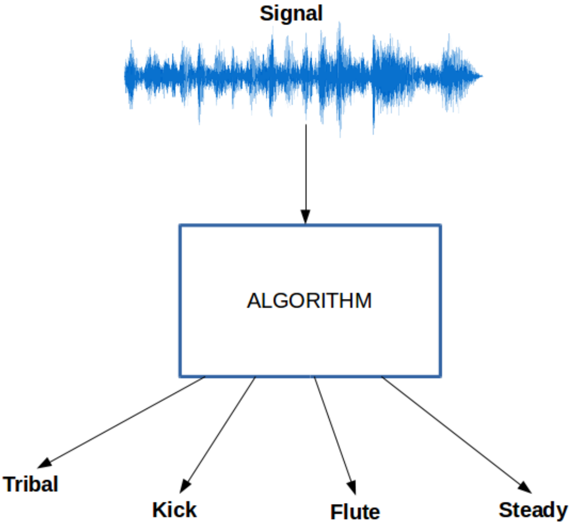
aravind.sankaran@rwth-aachen.de

April 7, 2017

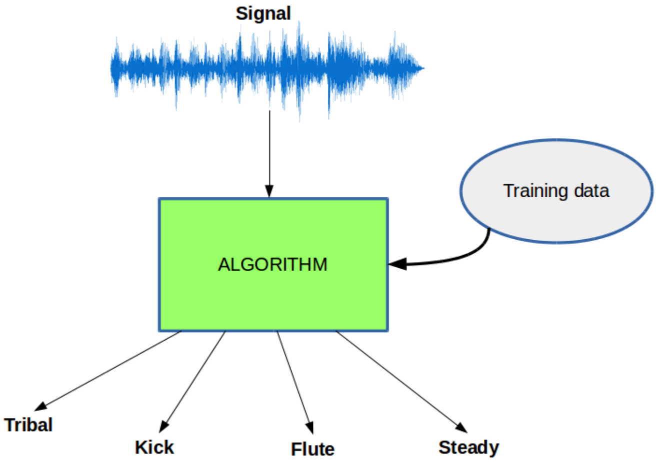
Acknowledgement

- Prof. Paolo Bientinesi
- Prof. Marco Alunno

AIM



AIM



User specific recommendation system?

→ **Do you have to create a dataset with ground-truth?**

User specific recommendation system?

→ **Do you have to create a dataset with ground-truth?**

→ *Lets look at some solutions where you* **don't** *have to..*

Collaborative filtering :

- Exploits social trends
- No information from audio content is used
- Cold-Start Problem

APPLICATION

User specific recommendation system?

$\rightarrow$ **Do you have to create a dataset with ground-truth?**

$\rightarrow$ *Lets look at some solutions where you* **don't** *have to..*

Collaborative + Content-based :

- Gather training data by crowd sourcing
- User specific recommendations by filtering popular tags

User specific recommendation system?

→ **When do you have to create a dataset with ground-truth?**

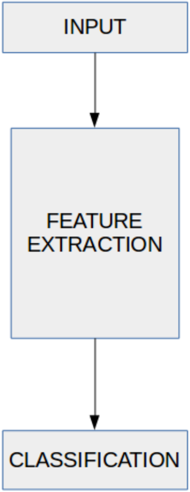
APPLICATION

Recommendation system for experts?

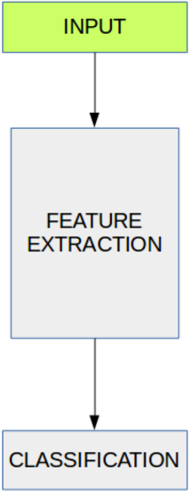
$\rightarrow$ **When do you have to create a dataset with ground-truth?**

$\rightarrow$ *Lets look from an artist's point of view ..*

PIPELINE



PIPELINE



Inputs :

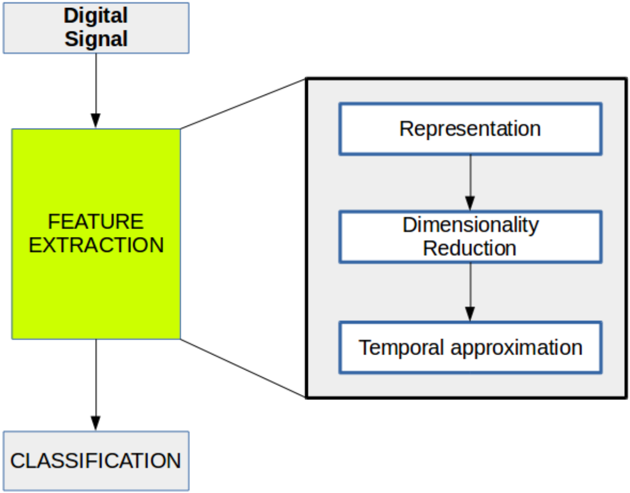
- Digital Signal (.mp3, .wav)
- Sheets of musical notes

PIPELINE

Feature Extraction :

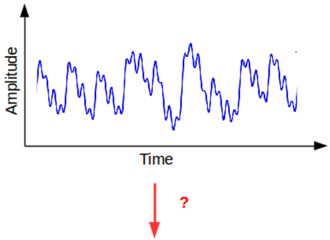
- $\mathbb{R}^N \rightarrow \mathbb{R}^T \quad T < N$
- **Organized :**
  Encode information about discriminants
- **Robust :**
  Transformation should be well posed

PIPELINE



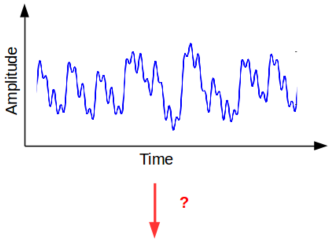
Digital Signal
FEATURE EXTRACTION
Representation
Dimensionality Reduction
Temporal approximation
CLASSIFICATION

REPRESENTATION



$$\mathbf{a} = (a_1, a_2, ..a_N) = a_1\mathbf{e}_1 + ...a_N\mathbf{e}_N$$

REPRESENTATION

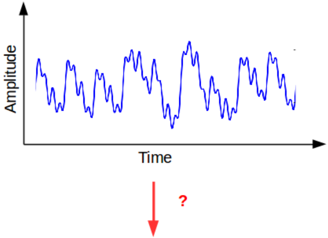


$$\mathbf{a} = (a_1, a_2, ..a_N) = a_1\mathbf{e}_1 + ...a_N\mathbf{e}_N$$

Basis

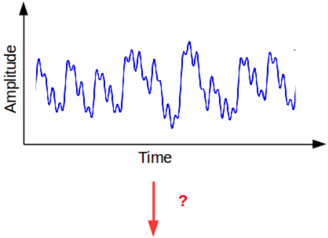
A group of vectors forms a basis of a vector space $\mathbb{V}$ if every vector in $\mathbb{V}$ can be represented as a linear combination of the basis vectors

REPRESENTATION



$$\mathbf{a} = a_1\mathbf{e}_1 + \dots a_N\mathbf{e}_N = \mathbb{1}\mathbf{a}$$

$$\mathbf{a} = c_1\mathbf{q}_1 + \dots c_M\mathbf{q}_M = \mathbf{Q}\mathbf{c}$$

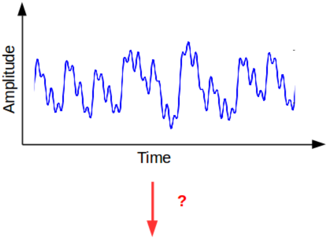


$$\mathbf{a} = a_1\mathbf{e}_1 + \ldots a_N\mathbf{e}_N = \mathbb{1}\mathbf{a}$$

$$\mathbf{a} = c_1\mathbf{q}_1 + \ldots c_M\mathbf{q}_M = \mathbf{Q}\mathbf{c}$$

$$\mathbf{Q}^{-1}\mathbf{a} = \mathbf{c} \qquad \mathbf{Q}^{-1} \in \mathbb{R}^{M \times N}$$

$$\mathbf{Q}^{-1}\mathbf{a} = \mathbf{c} \qquad \mathbf{Q}^{-1} \in \mathbb{R}^{M \times N}$$

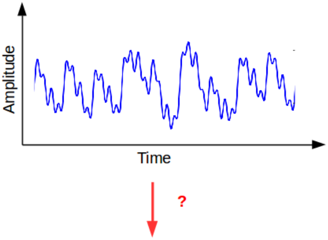


Exponential Fourier Theorem

Complex exponentials which are functions of frequencies form basis for *periodic* function

REPRESENTATION

$$\mathbf{Q}^{-1}\mathbf{a} = \mathbf{c} \qquad \mathbf{Q}^{-1} \in \mathbb{R}^{M \times N}$$



Exponential Fourier Theorem

Complex exponentials which are functions of frequencies form basis for *periodic* function

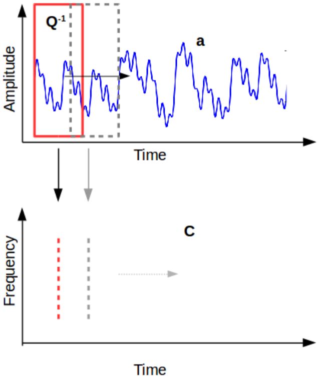
Fourier Transform

Application of *Fourier Theorem* for general signals.

$$\mathbf{Q}^{-1}[i] = \mathbf{e}^{-j\omega t} \qquad i \in \{0, 1.., M\}$$

REPRESENTATION

$$\mathbf{Q}^{-1}\mathbf{a} = \mathbf{c} \qquad \mathbf{Q}^{-1} \in \mathbb{R}^{M \times N}$$



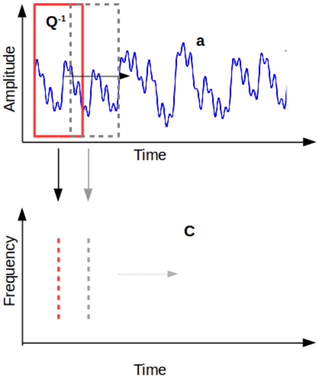
Fourier Transform

Application of *Fourier Theorem* for general signals.

$$\mathbf{Q}^{-1}[i] = \mathbf{e}^{-j\omega t} \qquad i \in \{0, 1.., M\}$$

REPRESENTATION

$$\mathbf{Q}^{-1}\mathbf{a} = \mathbf{c} \qquad \mathbf{Q}^{-1} \in \mathbb{R}^{M \times N}$$



Fourier Transform

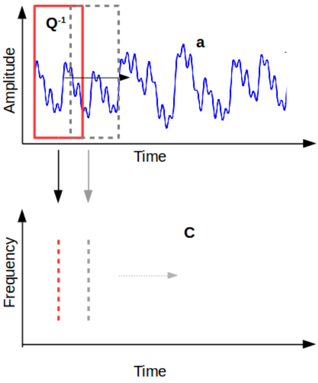
Application of *Fourier Theorem* for general signals.

$$\mathbf{Q}^{-1}[i] = \mathbf{e}^{-j\omega t} \qquad i \in \{0, 1.., M\}$$

Short-time Fourier Transform

$$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1} \qquad \mathbf{C} \in \mathbb{C}^{M \times P}$$

REPRESENTATION



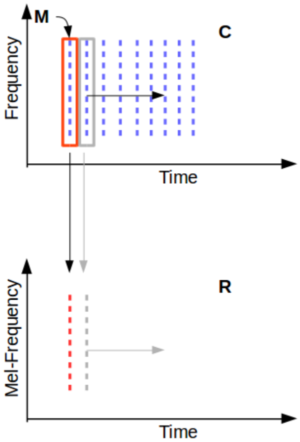
Short-time Fourier Transform

$$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1} \qquad \mathbf{C} \in \mathbb{C}^{M \times P}$$

Fast Fourier Transform

Faster version of STFT that exploits the symmetry of sinusoids.
STFT : $O(N^2)$
FFT : $O(NlogN)$

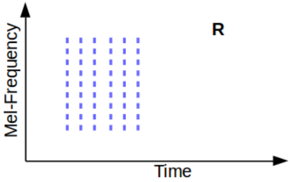
REPRESENTATION



$$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1} \qquad \mathbf{C} \in \mathbb{C}^{M \times P}$$

Spectrogram representations

- **Mel Spec :**
  $\mathbf{R} = \mathbf{M}.\mathbf{C} \quad \vee\, \mathbf{M} \in \mathbb{R}^{R \times M}$
- **Chromagram :**
  $\mathbf{R} = \mathbf{M}_C.\mathbf{C}$
- **Tempogram :**
  $\mathbf{R} = \mathbf{C} \star \mathbf{M}_T$



$$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1} \qquad \mathbf{C} \in \mathbb{C}^{M \times P}$$

$$\mathbf{R} = \mathbf{C} \star \mathbf{M} \qquad \mathbf{R} \in \mathbb{R}^{R \times P}$$

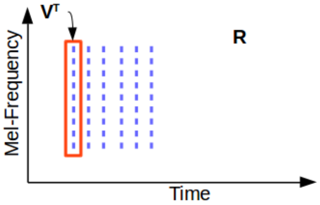
Principal Component Analysis

Represent **R** in a basis that is a function of variance in the information.

DIMENSIONALITY REDUCTION

$$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1} \qquad \mathbf{C} \in \mathbb{C}^{M \times P}$$

$$\mathbf{R} = \mathbf{C} \star \mathbf{M} \qquad \mathbf{R} \in \mathbb{R}^{R \times P}$$



Principal Component Analysis

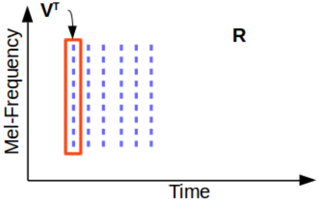
Represent **R** in a basis that is a function of variance in the information.

$\hat{\mathbf{R}} = \mathit{Center}(\mathbf{R})$

$\boldsymbol{\Sigma} = \frac{1}{P}\hat{\mathbf{R}}\hat{\mathbf{R}}^T$

$\mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^T = \boldsymbol{\Sigma}$

$\mathbf{X} = \mathit{Truncate}(\mathbf{V}^T)\hat{\mathbf{R}}$

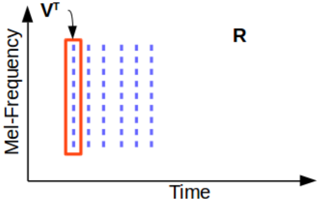


$$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1} \qquad \mathbf{C} \in \mathbb{C}^{M \times P}$$

$$\mathbf{R} = \mathbf{C} \star \mathbf{M} \qquad \mathbf{R} \in \mathbb{R}^{R \times P}$$

Mel-Frequency Cepstral Coefficients

Basis functions of *principal components* of log spectra are very similiar to *cosine transform*
$\mathbf{V}^T[i] = cos(\omega t) \quad i \in \{0, 1.., T\}$

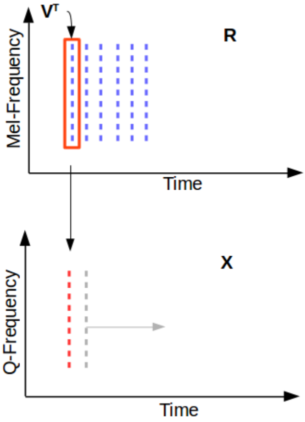


$$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1} \qquad \mathbf{C} \in \mathbb{C}^{M \times P}$$

$$\mathbf{R} = \mathbf{C} \star \mathbf{M} \qquad \mathbf{R} \in \mathbb{R}^{R \times P}$$

Mel-Frequency Cepstral Coefficients

Basis functions of *principal components* of log spectra are very similiar to *cosine transform*
$\mathbf{V}^T[i] = cos(\omega t) \quad i \in \{0, 1.., T\}$

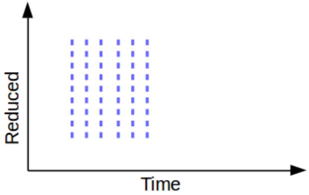


$$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1} \qquad \mathbf{C} \in \mathbb{C}^{M \times P}$$
$$\mathbf{R} = \mathbf{C} \star \mathbf{M} \qquad \mathbf{R} \in \mathbb{R}^{R \times P}$$
$$\mathbf{X} = \mathbf{R} \star \mathbf{V}^{T} \qquad \mathbf{X} \in \mathbb{R}^{T \times P}$$

Mel-Frequency Cepstral Coefficients

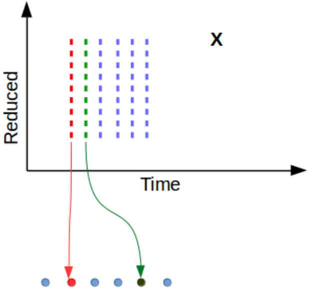
Basis functions of *principal components* of log spectra are very similiar to *cosine transform*
$\mathbf{V}^T[i] = cos(\omega t) \quad i \in \{0, 1.., T\}$

TEMPORAL APPROXIMATION



$$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1} \qquad \mathbf{C} \in \mathbb{C}^{M \times P}$$
$$\mathbf{R} = \mathbf{C} \star \mathbf{M} \qquad \mathbf{R} \in \mathbb{R}^{R \times P}$$
$$\mathbf{X} = \mathbf{R} \star \mathbf{V}^{T} \qquad \mathbf{X} \in \mathbb{R}^{T \times P}$$

TEMPORAL APPROXIMATION



$$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1} \qquad \mathbf{C} \in \mathbb{C}^{M \times P}$$
$$\mathbf{R} = \mathbf{C} \star \mathbf{M} \qquad \mathbf{R} \in \mathbb{R}^{R \times P}$$
$$\mathbf{X} = \mathbf{R} \star \mathbf{V}^{T} \qquad \mathbf{X} \in \mathbb{R}^{T \times P}$$

Bag Of Frames

- Assign each column of $\mathbf{X}$ to the nearest of K clusters.
- Count the number of assignments to each of the K clusters.
- The resulting feature is of dimension K

K - Means[1] :

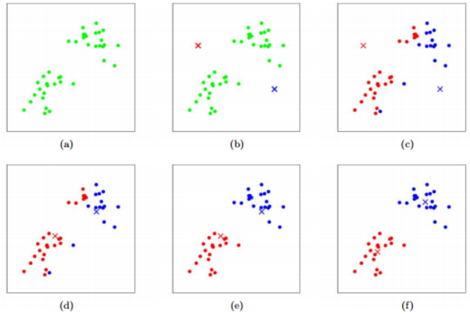


Figure 1: K-means algorithm. Training examples are shown as dots, and cluster centroids are shown as crosses. (a) Original dataset. (b) Random initial cluster centroids. (c-f) Illustration of running two iterations of k-means. In each iteration, we assign each training example to the closest cluster centroid (shown by "painting" the training examples the same color as the cluster centroid to which is assigned); then we move each cluster centroid to the mean of the points assigned to it. Images courtesy of Michael Jordan.

CLASSIFICATION

Feature Extraction : $\mathbb{R}^{N_f} \rightarrow \mathbb{R}^K$

$$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1} \qquad \mathbf{C} \in \mathbb{C}^{M \times P}$$
$$\mathbf{R} = \mathbf{C} \star \mathbf{M} \qquad \mathbf{R} \in \mathbb{R}^{R \times P}$$
$$\mathbf{X} = \mathbf{R} \star \mathbf{V}^T \qquad \mathbf{X} \in \mathbb{R}^{T \times P}$$
$$\mathbf{f} = \mathit{Temporal\_Approx}(\mathbf{X}) \qquad \mathbf{f} \in \mathbb{R}^K$$

Classification : $\mathbb{R}^K \rightarrow \mathbb{R}^L$

$$\mathbf{y} = \Phi(\mathbf{f}) \qquad \mathbf{y} \in \mathbb{R}^L$$

CLASSIFICATION

Single-layer perceptron :

$$\mathbf{y} = \mathbf{W}\mathbf{f} \qquad \mathbf{W} \in \mathbb{R}^{L \times K}$$

FE : $\mathbb{R}^{N_f} \rightarrow \mathbb{R}^{K}$

$$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1} \qquad \mathbf{C} \in \mathbb{C}^{M \times P}$$
$$\mathbf{R} = \mathbf{C} \star \mathbf{M} \qquad \mathbf{R} \in \mathbb{R}^{R \times P}$$
$$\mathbf{X} = \mathbf{R} \star \mathbf{V}^{T} \qquad \mathbf{X} \in \mathbb{R}^{T \times P}$$
$$\mathbf{f} = T(\mathbf{X}) \qquad \mathbf{f} \in \mathbb{R}^{K}$$

Classification : $\mathbb{R}^{K} \rightarrow \mathbb{R}^{L}$

$$\mathbf{y} = \Phi(\mathbf{f}) \qquad \mathbf{y} \in \mathbb{R}^{L}$$

CLASSIFICATION

Single-layer perceptron :

$$\mathbf{y} = \color{red}{\mathbf{W}}\color{black}\mathbf{f} \qquad \mathbf{W} \in \mathbb{R}^{L \times K}$$

Solve for $\color{red}{\mathbf{W}}$ with the training data

FE : $\mathbb{R}^{N_f} \rightarrow \mathbb{R}^{K}$

$$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1} \qquad \mathbf{C} \in \mathbb{C}^{M \times P}$$
$$\mathbf{R} = \mathbf{C} \star \mathbf{M} \qquad \mathbf{R} \in \mathbb{R}^{R \times P}$$
$$\mathbf{X} = \mathbf{R} \star \mathbf{V}^{T} \qquad \mathbf{X} \in \mathbb{R}^{T \times P}$$
$$\mathbf{f} = T(\mathbf{X}) \qquad \mathbf{f} \in \mathbb{R}^{K}$$

Classification : $\mathbb{R}^{K} \rightarrow \mathbb{R}^{L}$

$$\mathbf{y} = \Phi(\mathbf{f}) \qquad \mathbf{y} \in \mathbb{R}^{L}$$

Two-layer perceptron :

$$\mathbf{y} = \sigma(\mathbf{W}_2 ReLU(\mathbf{W}_1 \mathbf{f}))$$

$$\mathbf{W}_2 \in \mathbb{R}^{L \times H} \quad \mathbf{W}_1 \in \mathbb{R}^{H \times K}$$

FE : $\mathbb{R}^{N_f} \to \mathbb{R}^K$

$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1} \qquad \mathbf{C} \in \mathbb{C}^{M \times P}$

$\mathbf{R} = \mathbf{C} \star \mathbf{M} \qquad \mathbf{R} \in \mathbb{R}^{R \times P}$

$\mathbf{X} = \mathbf{R} \star \mathbf{V}^T \qquad \mathbf{X} \in \mathbb{R}^{T \times P}$

$\mathbf{f} = T(\mathbf{X}) \qquad \mathbf{f} \in \mathbb{R}^K$

Classification : $\mathbb{R}^K \to \mathbb{R}^L$

$\mathbf{y} = \Phi(\mathbf{f}) \qquad \mathbf{y} \in \mathbb{R}^L$

$$\mathbf{y} = \sigma(\mathbf{W}_2 ReLU(\mathbf{W}_1 \mathbf{f}))$$

Training :

- $E = loss(\mathbf{y}, \mathbf{t})$

FE : $\mathbb{R}^{N_f} \to \mathbb{R}^K$

$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1}$ $\quad \mathbf{C} \in \mathbb{C}^{M \times P}$

$\mathbf{R} = \mathbf{C} \star \mathbf{M}$ $\quad \mathbf{R} \in \mathbb{R}^{R \times P}$

$\mathbf{X} = \mathbf{R} \star \mathbf{V}^T$ $\quad \mathbf{X} \in \mathbb{R}^{T \times P}$

$\mathbf{f} = T(\mathbf{X})$ $\quad \mathbf{f} \in \mathbb{R}^K$

Classification : $\mathbb{R}^K \to \mathbb{R}^L$

$\mathbf{y} = \sigma(\mathbf{W}_2 ReLU(\mathbf{W}_1 \mathbf{f}))$ $\quad \mathbf{y} \in \mathbb{R}^L$

$$\mathbf{y} = \sigma(\mathbf{W}_2 ReLU(\mathbf{W}_1\mathbf{f}))$$

Training :

- $E = loss(\mathbf{y}, \mathbf{t})$
- $\frac{\partial E}{\partial \mathbf{W}_2} = \frac{\partial E}{\partial \mathbf{y}} \frac{\partial \mathbf{y}}{\partial \sigma} \frac{\partial \sigma}{\partial \mathbf{W}_2}$
- $\frac{\partial E}{\partial \mathbf{W}_1} = \frac{\partial E}{\partial \mathbf{y}} \frac{\partial \mathbf{y}}{\partial \sigma} \frac{\partial \sigma}{\partial ReLU} \frac{\partial ReLU}{\partial \mathbf{W}_1}$

FE : $\mathbb{R}^{N_f} \rightarrow \mathbb{R}^K$

$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1}$ $\quad \mathbf{C} \in \mathbb{C}^{M \times P}$

$\mathbf{R} = \mathbf{C} \star \mathbf{M}$ $\quad \mathbf{R} \in \mathbb{R}^{R \times P}$

$\mathbf{X} = \mathbf{R} \star \mathbf{V}^T$ $\quad \mathbf{X} \in \mathbb{R}^{T \times P}$

$\mathbf{f} = T(\mathbf{X})$ $\quad \mathbf{f} \in \mathbb{R}^K$

Classification : $\mathbb{R}^K \rightarrow \mathbb{R}^L$

$\mathbf{y} = \sigma(\mathbf{W}_2 ReLU(\mathbf{W}_1\mathbf{f}))$ $\quad \mathbf{y} \in \mathbb{R}^L$

$$\mathbf{y} = \sigma(\mathbf{W}_2 ReLU(\mathbf{W}_1 \mathbf{f}))$$

Training :

- $E = loss(\mathbf{y}, \mathbf{t})$
- $\frac{\partial E}{\partial \mathbf{W}_2} = \frac{\partial E}{\partial \mathbf{y}} \frac{\partial \mathbf{y}}{\partial \sigma} \frac{\partial \sigma}{\partial \mathbf{W}_2}$
- $\frac{\partial E}{\partial \mathbf{W}_1} = \frac{\partial E}{\partial \mathbf{y}} \frac{\partial \mathbf{y}}{\partial \sigma} \frac{\partial \sigma}{\partial ReLU} \frac{\partial ReLU}{\partial \mathbf{W}_1}$
- $\mathbf{W}_1 \leftarrow update(\mathbf{W}_1, \frac{\partial E}{\partial \mathbf{W}_1})$
- $\mathbf{W}_2 \leftarrow update(\mathbf{W}_2, \frac{\partial E}{\partial \mathbf{W}_2})$

FE : $\mathbb{R}^{N_f} \rightarrow \mathbb{R}^K$

$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1}$ $\quad \mathbf{C} \in \mathbb{C}^{M \times P}$
$\mathbf{R} = \mathbf{C} \star \mathbf{M}$ $\quad \mathbf{R} \in \mathbb{R}^{R \times P}$
$\mathbf{X} = \mathbf{R} \star \mathbf{V}^T$ $\quad \mathbf{X} \in \mathbb{R}^{T \times P}$
$\mathbf{f} = T(\mathbf{X})$ $\quad \mathbf{f} \in \mathbb{R}^K$

Classification : $\mathbb{R}^K \rightarrow \mathbb{R}^L$

$\mathbf{y} = \sigma(\mathbf{W}_2 ReLU(\mathbf{W}_1 \mathbf{f}))$ $\quad \mathbf{y} \in \mathbb{R}^L$

OVERVIEW

FE : $\mathbb{R}^{N_f} \rightarrow \mathbb{R}^K$

$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1} \qquad \mathbf{C} \in \mathbb{C}^{M \times P}$

$\mathbf{R} = \mathbf{C} \star \mathbf{M} \qquad \mathbf{R} \in \mathbb{R}^{R \times P}$

$\mathbf{X} = \mathbf{R} \star \mathbf{V}^T \qquad \mathbf{X} \in \mathbb{R}^{T \times P}$

$\mathbf{f} = T(\mathbf{X}) \qquad \mathbf{f} \in \mathbb{R}^K$

Classification : $\mathbb{R}^K \rightarrow \mathbb{R}^L$

$\mathbf{y} = \sigma(\mathbf{W}_2 ReLU(\mathbf{W}_1 \mathbf{f})) \quad \mathbf{y} \in \mathbb{R}^L$

Recurrent Neural Network:

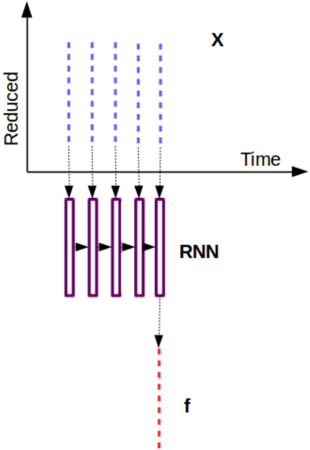
FE : $\mathbb{R}^{N_f} \rightarrow \mathbb{R}^K$

$$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1} \qquad \mathbf{C} \in \mathbb{C}^{M \times P}$$
$$\mathbf{R} = \mathbf{C} \star \mathbf{M} \qquad \mathbf{R} \in \mathbb{R}^{R \times P}$$
$$\mathbf{X} = \mathbf{R} \star \mathbf{V}^T \qquad \mathbf{X} \in \mathbb{R}^{T \times P}$$
$$\mathbf{f} = T(\mathbf{X}) \qquad \mathbf{f} \in \mathbb{R}^K$$

Classification : $\mathbb{R}^K \rightarrow \mathbb{R}^L$

$$\mathbf{y} = \sigma(\mathbf{W}_2 ReLU(\mathbf{W}_1 \mathbf{f})) \quad \mathbf{y} \in \mathbb{R}^L$$

Recurrent Neural Network:
(Sequence to One)



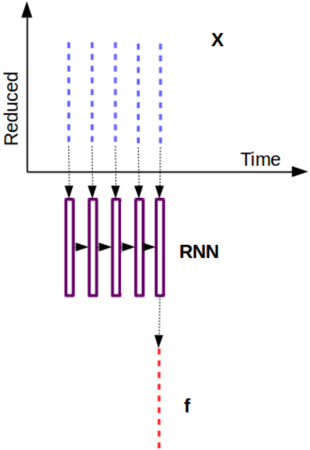
FE : $\mathbb{R}^{N_f} \rightarrow \mathbb{R}^K$

$$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1} \qquad \mathbf{C} \in \mathbb{C}^{M \times P}$$
$$\mathbf{R} = \mathbf{C} \star \mathbf{M} \qquad \mathbf{R} \in \mathbb{R}^{R \times P}$$
$$\mathbf{X} = \mathbf{R} \star \mathbf{V}^T \qquad \mathbf{X} \in \mathbb{R}^{T \times P}$$
$$\mathbf{f} = T(\mathbf{X}) \qquad \mathbf{f} \in \mathbb{R}^K$$

Classification : $\mathbb{R}^K \rightarrow \mathbb{R}^L$

$$\mathbf{y} = \sigma(\mathbf{W}_2 ReLU(\mathbf{W}_1 \mathbf{f})) \qquad \mathbf{y} \in \mathbb{R}^L$$

Recurrent Neural Network:
(Sequence to One)



RNN

$\mathbf{f} = \mathbf{W}_3\mathbf{h}_P$
$\mathbf{h}_P = \Phi(\mathbf{X}[P], \mathbf{X}[P-1], ..\mathbf{X}[0])$

SUPERVISING FEATURES

Recurrent Neural Network:
(Sequence to One)

RNN

$\mathbf{f} = \mathbf{W}_3\mathbf{h}_P \qquad \mathbf{h}_P = \Phi(\mathbf{X}[P], \mathbf{X}[P-1], ..\mathbf{X}[0])$

Long Short Term Memory

$$\mathbf{f} = \mathbf{W}_3\mathbf{h}_P$$
$$\mathbf{h}_p = \mathbf{o}_p \odot \sigma_h(\mathbf{c}_p)$$
$$\mathbf{c}_p = \mathbf{g}_p \odot \mathbf{c}_{p-1} + \mathbf{i}_p \odot \sigma_c(\mathbf{W}_c\mathbf{x}_p + \mathbf{U}_c\mathbf{h}_{p-1})$$
$$\mathbf{o}_p = \sigma(\mathbf{W}_o\mathbf{x}_p + \mathbf{U}_o\mathbf{h}_{p-1})$$
$$\mathbf{i}_p = \sigma(\mathbf{W}_i\mathbf{x}_p + \mathbf{U}_i\mathbf{h}_{p-1})$$
$$\mathbf{g}_p = \sigma(\mathbf{W}_g\mathbf{x}_p + \mathbf{U}_g\mathbf{h}_{p-1})$$

SUPERVISING FEATURE

FE : $\mathbb{R}^{N_f} \rightarrow \mathbb{R}^K$

$$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1} \qquad \mathbf{C} \in \mathbb{C}^{M \times P}$$
$$\mathbf{R} = \mathbf{C} \star \mathbf{M} \qquad \mathbf{R} \in \mathbb{R}^{R \times P}$$
$$\mathbf{X} = \mathbf{R} \star \mathbf{V}^T \qquad \mathbf{X} \in \mathbb{R}^{T \times P}$$
$$\mathbf{f} = LSTM(\mathbf{X}) \qquad \mathbf{f} \in \mathbb{R}^K$$

Classification : $\mathbb{R}^K \rightarrow \mathbb{R}^L$

$$\mathbf{y} = \sigma(\mathbf{W}_2 ReLU(\mathbf{W}_1 \mathbf{f})) \qquad \mathbf{y} \in \mathbb{R}^L$$

Convolution Neural Network :

FE : $\mathbb{R}^{N_f} \rightarrow \mathbb{R}^{K}$

$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1} \qquad \mathbf{C} \in \mathbb{C}^{M \times P}$

$\mathbf{R} = \mathbf{C} \star \mathbf{M} \qquad \mathbf{R} \in \mathbb{R}^{R \times P}$

$\mathbf{X} = \mathbf{R} \star \mathbf{W}_4 \qquad \mathbf{X} \in \mathbb{R}^{T \times P}$

$\mathbf{f} = LSTM(\mathbf{X}) \qquad \mathbf{f} \in \mathbb{R}^{K}$

Classification : $\mathbb{R}^{K} \rightarrow \mathbb{R}^{L}$

$\mathbf{y} = \sigma(\mathbf{W}_2 ReLU(\mathbf{W}_1 \mathbf{f})) \quad \mathbf{y} \in \mathbb{R}^{L}$

Convolution Neural Network :

FE : $\mathbb{R}^{N_f} \rightarrow \mathbb{R}^K$

$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1} \qquad \mathbf{C} \in \mathbb{C}^{M \times P}$

$\mathbf{R} = \mathbf{C} \star \mathbf{M} \qquad \mathbf{R} \in \mathbb{R}^{R \times P}$

$\mathbf{X}_1 = \Phi(\mathbf{R} \star \mathbf{W}_6)$

$\mathbf{X}_2 = \Phi(\mathbf{X}_1 \star \mathbf{W}_5)$

$\mathbf{X} = \Phi(\mathbf{X}_2 \star \mathbf{W}_4) \qquad \mathbf{X} \in \mathbb{R}^{T \times P}$

$\mathbf{f} = LSTM(\mathbf{X}) \qquad \mathbf{f} \in \mathbb{R}^K$

Classification : $\mathbb{R}^K \rightarrow \mathbb{R}^L$

$\mathbf{y} = \sigma(\mathbf{W}_2 ReLU(\mathbf{W}_1 \mathbf{f})) \quad \mathbf{y} \in \mathbb{R}^L$

SUPERVISING FEATURE - Deep learning

Deep Learning issues :

- Vanishing gradients
- Need large amount of training data

Solutions :

- Φ : Non-linearities, Drop out, Batch Normalization
- Transfer learning (Black-box / Fine tune)

FE : $\mathbb{R}^{N_f} \to \mathbb{R}^K$

$$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1} \qquad \mathbf{C} \in \mathbb{C}^{M \times P}$$
$$\mathbf{R} = \mathbf{C} \star \mathbf{M} \qquad \mathbf{R} \in \mathbb{R}^{R \times P}$$
$$\mathbf{X}_1 = \Phi(\mathbf{R} \star \mathbf{W}_6)$$
$$\mathbf{X}_2 = \Phi(\mathbf{X}_1 \star \mathbf{W}_5)$$
$$\mathbf{X} = \Phi(\mathbf{X}_2 \star \mathbf{W}_4) \qquad \mathbf{X} \in \mathbb{R}^{T \times P}$$
$$\mathbf{f} = LSTM(\mathbf{X}) \qquad \mathbf{f} \in \mathbb{R}^K$$

Classification : $\mathbb{R}^K \to \mathbb{R}^L$

$$\mathbf{y} = \sigma(\mathbf{W}_2 ReLU(\mathbf{W}_1 \mathbf{f})) \qquad \mathbf{y} \in \mathbb{R}^L$$

Experiments

Choi's CNN[2] + BoF :
AUC 0.67

$$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1} \quad \mathbf{C} \in \mathbb{C}^{M \times P}$$
$$\mathbf{R} = \mathbf{C} \star \mathbf{M} \quad \mathbf{R} \in \mathbb{R}^{96 \times P}$$
$$\mathbf{X} = Cnn5(\mathbf{R}) \quad \mathbf{X} \in \mathbb{R}^{1366 \times W}$$
$$\mathbf{f} = BoF(\mathbf{X}) \quad \mathbf{f} \in \mathbb{R}^{1024}$$

Choi's CNN[2] + LSTM :
AUC 0.71

$$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1} \quad \mathbf{C} \in \mathbb{C}^{M \times P}$$
$$\mathbf{R} = \mathbf{C} \star \mathbf{M} \quad \mathbf{R} \in \mathbb{R}^{96 \times P}$$
$$\mathbf{X} = Cnn5(\mathbf{R}) \quad \mathbf{X} \in \mathbb{R}^{1366 \times W}$$
$$\mathbf{f} = LSTM\_2(\mathbf{X}) \quad \mathbf{f} \in \mathbb{R}^{1024}$$

MFCC + BoF :
AUC 0.62

$$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1} \quad \mathbf{C} \in \mathbb{C}^{M \times P}$$
$$\mathbf{R} = \mathbf{C} \star \mathbf{M} \quad \mathbf{R} \in \mathbb{R}^{96 \times P}$$
$$\mathbf{X} = \mathbf{R} \star \mathbf{V}^T \quad \mathbf{X} \in \mathbb{R}^{90 \times P}$$
$$\mathbf{f} = BoF(\mathbf{X}) \quad \mathbf{f} \in \mathbb{R}^{1024}$$

MFCC + LSTM :
AUC **0.74**

$$\mathbf{C} = \mathbf{a} \star \mathbf{Q}^{-1} \quad \mathbf{C} \in \mathbb{C}^{M \times P}$$
$$\mathbf{R} = \mathbf{C} \star \mathbf{M} \quad \mathbf{R} \in \mathbb{R}^{96 \times P}$$
$$\mathbf{X} = \mathbf{R} \star \mathbf{V}^T \quad \mathbf{X} \in \mathbb{R}^{90 \times P}$$
$$\mathbf{f} = LSTM\_2(\mathbf{X}) \quad \mathbf{f} \in \mathbb{R}^{1024}$$

Conclusion

First teach the algorithm to decompose the rhythmic traces?