Advances in Music Information Retrieval using Deep Learning Techniques

- Sid Pramod
Music Information Retrieval (MIR)

- Science of retrieving information from music.
- Includes tasks such as
  - Query by Example, Query by Humming
  - Music Recommendation, Automatic Playlist Generation
  - Genre Classification, Artist Classification, Instrument Classification, Chord recognition
  - Emotion recognition
Solutions

- Commonly use Machine Learning (ML)

Raw Data (Audio Signal) → Feature Extraction (representation) → Machine Learning Algorithm (e.g. SVM, kNN, etc.)
Feature Design

- Feature extraction stage usually involves hand-crafted feature design
  - ML Algorithm performance critically depends on feature design
  - Features have to be robust to noise, translations and other variations
  - Hand-crafted features are heuristic based
  - Most common features are Mel-frequency Cepstral Coefficients (MFCCs) and variants
Learning Features/Representations

● Learning representations is far less tedious than engineering features
● Context-dependent feature extraction is made possible
● Not necessarily task-specific, “Transfer learning” allows reusing features across multiple tasks
Learning Features/Representations

• Vector Quantization
  ○ Use simple feature extraction techniques
  ○ Perform clustering on all data points
  ○ Transformed representation is a vector where a single entry is non-zero, whose index corresponds to cluster id.

  e.g. \([x_1, x_2, x_3 \ldots x_n] \rightarrow [0, 0, \ldots 1, \ldots 0]\)
Learning Features/Representations

- **Sparse Coding**
  - A class of algorithms that learn to represent each data point as a linear combination of basis vectors (features)
  - Set of basis vectors form the “dictionary”
  
  e.g. if \([x_1, x_2, x_3 \ldots x_n] = a_1.f_1 + 0.f_2 + 0.f_3 + a_4.f_4 + \ldots\)
  then \([x_1, x_2, x_3 \ldots x_n] \rightarrow [a_1, 0, 0, a_4, \ldots]\)
Learning Features/Representations

- Autoencoder
  - A feed forward neural network with one hidden layer and same number of output nodes as input
  - Task is to reconstruct the input
  - Hidden layer “learns” a sparse encoding of the data when constraints are placed on hidden layer activations during the training procedure
  - May be combined with supervised criterion
Input $\rightarrow$ Hidden Layer / Encoding $\rightarrow$ Reconstruction
Deep Architectures

● A stack of shallow transformations
  ○ Output of one stage serves as input to next
  ○ Complex transformation thus modeled as a series of simpler transformations
  ○ Each transformation encodes some specific variance

● Can be learned
  ○ Stacked auto-encoders and variants (unsupervised)
  ○ Deep feedforward neural networks (supervised)
Does it work?

- Humphrey et al. [2012] review initial research in the area
  - All of which achieve state of the art performance
  - Tasks include genre recognition, instrument classification, chord recognition, ...
Does it work?

- Musical Onset Detection using CNNs - Schlüter et al. [2014]

<table>
<thead>
<tr>
<th>Model</th>
<th>Precision</th>
<th>Recall</th>
<th>F-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>RNN [10, 5]</td>
<td>0.892</td>
<td>0.855</td>
<td>0.873</td>
</tr>
<tr>
<td>CNN [1]</td>
<td>0.905</td>
<td>0.866</td>
<td>0.885</td>
</tr>
<tr>
<td>+ Dropout</td>
<td>0.909</td>
<td>0.871</td>
<td>0.890</td>
</tr>
<tr>
<td>+ Fuzziness</td>
<td>0.914</td>
<td>0.885</td>
<td>0.899</td>
</tr>
<tr>
<td>+ ReLU</td>
<td>0.917</td>
<td>0.889</td>
<td>0.903</td>
</tr>
<tr>
<td>SuperFlux [5]</td>
<td>0.883</td>
<td>0.793</td>
<td>0.836</td>
</tr>
</tbody>
</table>
Does it work?

- Deep content-based music recommendation - Oord et al. [2013]

<table>
<thead>
<tr>
<th>Model</th>
<th>mAP</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLR</td>
<td>0.01801</td>
<td>0.60608</td>
</tr>
<tr>
<td>linear regression</td>
<td>0.02389</td>
<td>0.63518</td>
</tr>
<tr>
<td>MLP</td>
<td>0.02536</td>
<td>0.64611</td>
</tr>
<tr>
<td>CNN with MSE</td>
<td>0.05016</td>
<td>0.70987</td>
</tr>
<tr>
<td>CNN with WPE</td>
<td>0.04323</td>
<td>0.70101</td>
</tr>
</tbody>
</table>

Table 2: Results for all considered models on a subset of the dataset containing only the 9,330 most popular songs, and listening data for 20,000 users.

<table>
<thead>
<tr>
<th>Model</th>
<th>mAP</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>random</td>
<td>0.00015</td>
<td>0.49935</td>
</tr>
<tr>
<td>linear regression</td>
<td>0.00101</td>
<td>0.64522</td>
</tr>
<tr>
<td>CNN with MSE</td>
<td>0.00672</td>
<td>0.77192</td>
</tr>
<tr>
<td>upper bound</td>
<td>0.23278</td>
<td>0.96070</td>
</tr>
</tbody>
</table>

Table 3: Results for linear regression on a bag-of-words representation of the audio signals, and a convolutional neural network trained with the MSE objective, on the full dataset (382,410 songs and 1 million users). Also shown are the scores achieved when the latent factor vectors are randomized, and when they are learned from usage data using WMF (upper bound).
References


