Music Genre Classification

Automatic Speech Recognition (CS 753)

Kalpesh | Chandrakanth | Rishabh

Task Definition

- To perform the task of genre classification of songs using existing classification techniques

- Stages involved in the task
  - Extract features from the song clip
  - Train classifier

Dataset

- GTZAN Dataset of 1000 songs of 30 seconds duration pertaining to 10 different genres

- Used 400 songs of 4 different genres

- Jazz, Pop, Metal and Classical

- Used 80% data for training and 20% data for testing

Approach

- Extracted the MFCC features for each frame of the songs
- GPPS features
- Classifiers

MFCC features based

- VQ
- CNN
- DNN using context frames

GPPS features based

- NN
- SVM

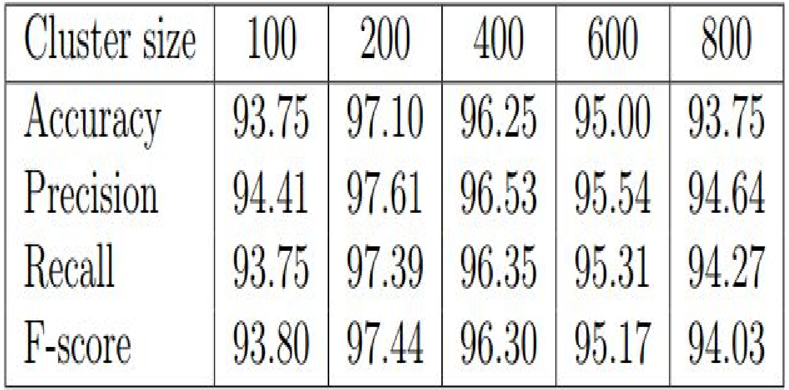
Vector Quantization

- Codebook generation for each genre
  - MFCC features of each frame of each genre as input
  - MiniBatchKMeans for clustering

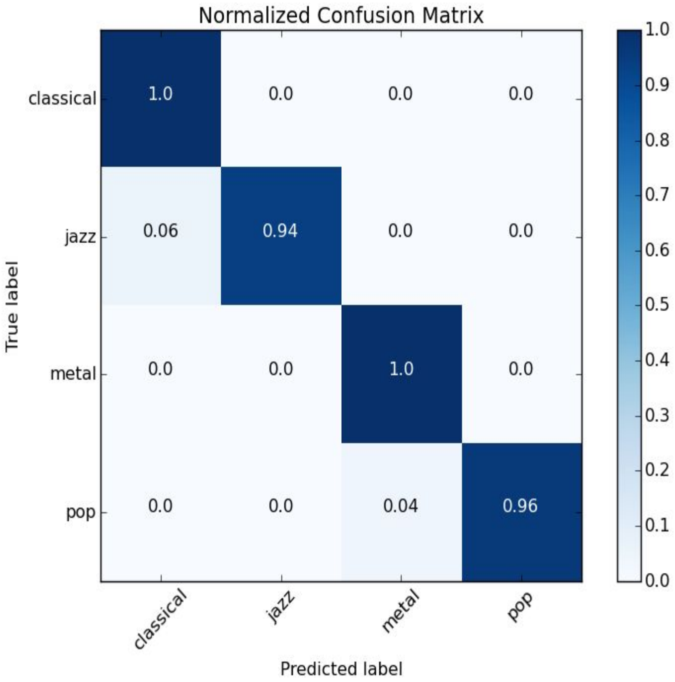
- A test song is classified based on the average distance of its frames from the nearest clusters of each genre

- Tried different number of clusters for each genre

Vector Quantization Results



Cluster size	100	200	400	600	800
Accuracy	93.75	97.10	96.25	95.00	93.75
Precision	94.41	97.61	96.53	95.54	94.64
Recall	93.75	97.39	96.35	95.31	94.27
F-score	93.80	97.44	96.30	95.17	94.03



Normalized Confusion Matrix

True label \ Predicted label	classical	jazz	metal	pop
classical	1.0	0.0	0.0	0.0
jazz	0.06	0.94	0.0	0.0
metal	0.0	0.0	1.0	0.0
pop	0.0	0.0	0.04	0.96

GPPS

Gaussian Posterior Probability Supervector

- GMM-UBM training
  - GMM trained on MFCC features of entire training data
  - Diagonal covariance matrices for reduced no. of parameters

- GPPS extraction

$$Pr(o_t|\lambda) = \sum_{j=1}^{J} w_j Pr(o_t|\mu_j, \Sigma_j)$$

$$\lambda = \{w_j, \mu_j, \Sigma_j\}, j = 1, 2..J$$

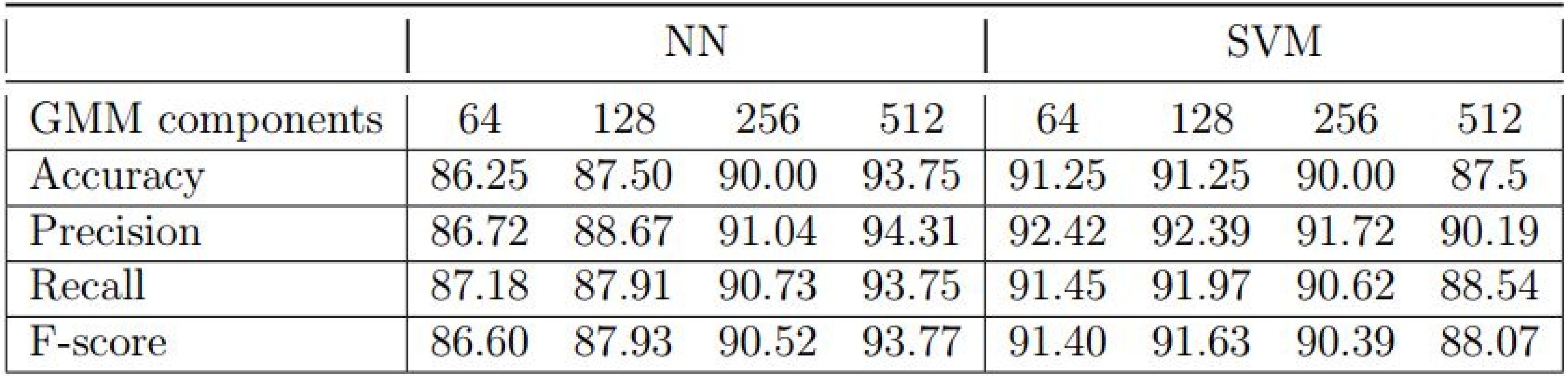
$$\kappa_j = \frac{1}{T} \sum_{t=1}^{T} \frac{w_j Pr(o_t|\mu_j, \Sigma_j)}{\sum_{j=1}^{J} w_j Pr(o_t|\mu_j, \Sigma_j)}$$

$$\kappa = [\kappa_1, \kappa_2, ...\kappa_J]$$

GPPS

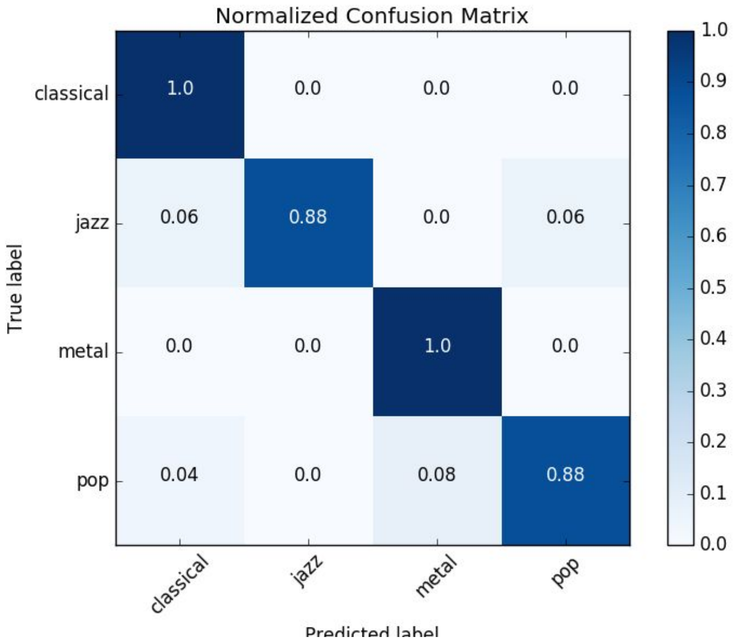
- Classifier
  - Input: GPPS vector of a song
    Output: Genre of that song
  - Possible choices
    - Neural networks
      InputLayer(J), Dense(10,relu), Dropout(0.5), Dense(4,softmax)
    - SVM
      Kernel - radial basis function
      Optimal parameters found - (C = 10000.0, gamma = 0.1)

GPPS results

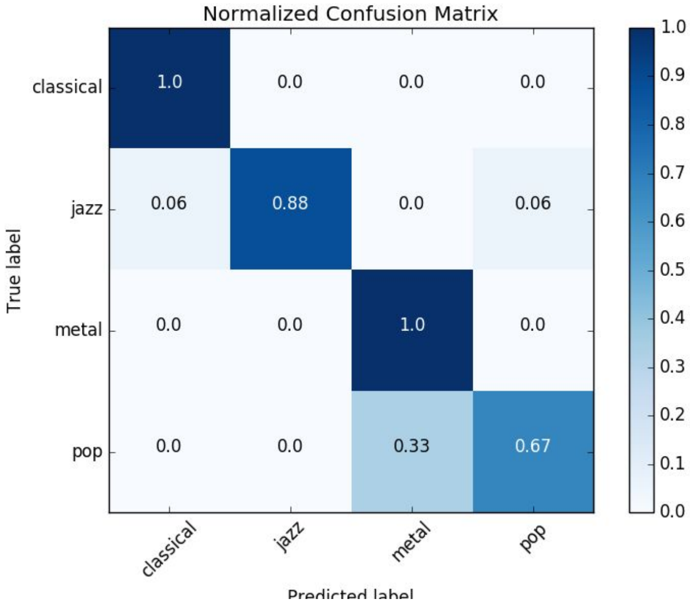


	NN				SVM			
GMM components	64	128	256	512	64	128	256	512
Accuracy	86.25	87.50	90.00	93.75	91.25	91.25	90.00	87.5
Precision	86.72	88.67	91.04	94.31	92.42	92.39	91.72	90.19
Recall	87.18	87.91	90.73	93.75	91.45	91.97	90.62	88.54
F-score	86.60	87.93	90.52	93.77	91.40	91.63	90.39	88.07

GPPS results



NN confusion matrix



SVM confusion matrix

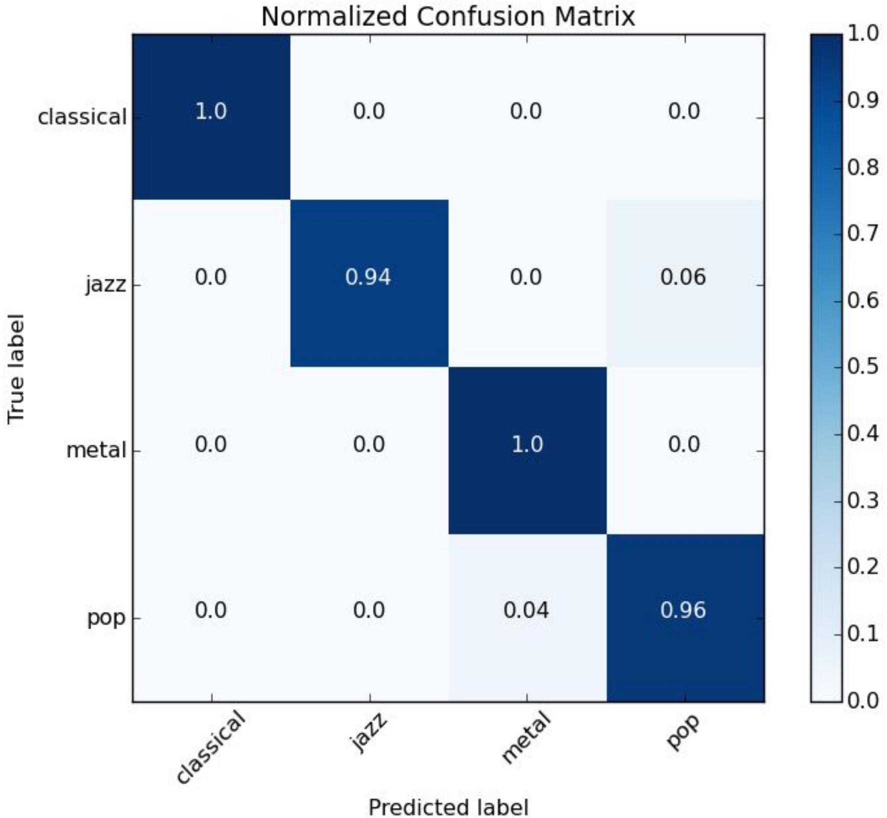
CNN

- Weight sharing and local connectivity

- Used a 1D CNN where local connectivity is across time

- Architecture
  - Input layer(2998*39) -> Conv1D(64,4,tanh) -> Maxpool1D(2) -> Conv1D(128,4,tanh) - > Maxpool1D(2) -> Conv1D(256,4,tanh) -> Dense(512) -> Dense(4)

CNN Results



Accuracy	97.50
Precision	97.77
Recall	97.40
F-score	97.54

DNN using context

- Used in classification tasks with highly non-linear class boundaries

- Context input frame

- Song classified using the maximum likelihood criterion

- Architecture
  - InputLayer(39*(2*5+1)) -> Dense(2650,relu) -> Dense(2650,relu) -> Dense(2650,relu) -> Dense(4)
  - Dropout(0.5)
- Achieved an accuracy of 82.5%

Conclusion and Future Work

- 1D-CNN and VQ have been found to be the best performing methods amongst the ones which were tried

- Most confusing genre pairs consistent across all methods.
  - True label: pop, predicted label: metal
    True label: jazz, predicted label: metal
    True label: jazz, predicted label: metal
  - Small set of songs get misclassified across all classifiers tried
  - Such songs get misclassified to the same genre across all classifiers

- Future Work
  - Try the classifiers for all the ten genres
  - Extend the project to map songs to music bands