11-775 Large Scale Multimedia Analysis
Spring 2019
HW 1

Sheetal Shalini
Andrew ID : sshalini

Collaboration Statement

1. Did you receive any help whatsoever from anyone in solving this assignment? Whiteboard discussion with Prashant Gupta, Shefali Garg, Karan Saxena, Ashwin Srinivasan
2. Did you give any help whatsoever to anyone in solving this assignment? Whiteboard discussion with Prashant Gupta, Shefali Garg, Karan Saxena, Ashwin Srinivasan
3. Did you find or come across code that implements any part of this assignment ? No

Problem Statement

The task is to perform multimedia event detection (MED) with audio features like MFCC and ASR transcripts.

Dataset

The dataset contains 2935 videos, with 3 positive events (P001: assembling shelter; P002: batting in run; P003: making cake) and 1 negative event class (NULL). There are 1002 training instances and 234 validation instances. The validation set is used to tune hyper-parameters, conduct ablation studies and report the average precision scores. There are 1699 test videos, for which the results have been predicted by the designed model and their accuracies have been compared.

Preprocessing

MFCC features

From each video, 40% of the frames have been randomly selected. Each frame is represented by an MFCC vector, which is of a fixed dimension. Mini Batch KMeans clustering has been performed on these frames in order to put the frames into 400 clusters based on their MFCC

vectors. Each video is then represented as a list of length 400, where each element represents the normalized count of the no.of frames of that video that belong to the respective index cluster. This vector of length 400 is the final feature vector of a video, passed on for training. Multiple classifiers have been trained on the feature vectors and their results are reported below.

ASR features

The text in each of the ASR files has been preprocessed, namely removal of punctuations, tokenizing into words and converting each word to lower case. Stopwords have been identified as the words with frequency greater than 800, and rare words have been identified as words with frequency = 1. Both, stopwords and rare words have been removed. The final vocabulary obtained consists of 5609 unique words. The ASR feature vectors are then created in a similar manner as MFCC vectors, and are then trained with multiple classifiers as reported below.

MED Pipeline

The following steps have been performed in the respective files mentioned:

1. **create_asr_vocab.py** : ASR vocabulary of 5609 unique words is created.
2. **create_asrfeat.py** : ASR features are extracted, preprocessed and their feature vectors are created.
3. **select_frames.py** : From each video, a certain ratio of the frames are selected at random to constitute the MFCC feature vectors.
4. **train_kmeans.py** : Mini Batch KMeans is trained to cluster the frames into 400 clusters according to their MFCC vectors.
5. **create_kmeans.py** : The MFCC feature vectors for each video are created, wherein each element represents the normalized count of the no.of frames from that video that appear in the respective index cluster.
6. **train_svm.py** : The classifiers are trained in this file.
7. **test_svm.py** : The predictions of the classifier models on the test set are generated.
8. **generate_submission.py** : This file compares the values in the 3 files generated by test_svm.py and generates the final submission csv.

Experiments and Results

The following different classifiers have been experimented with their default configurations, and their Average Precision (AP) on the validation set along with their accuracies on the test set with both the types of extracted features (MFCC and ASR) have been reported in the tables below.

Classifier	AP-P001	AP-P002	AP-P003	Accuracy
SVM	0.191354	**0.744239**	0.228128	0.41104
KNN	0.190192	0.196241	0.210588	0.55828
AdaBoost	0.23791	0.304359	0.223344	**0.62576**
Gradient Boosting	**0.535681**	0.427046	**0.294639**	0.56441
Gaussian Naive Bayes	0.198562	0.358651	0.235335	0.61963
Decision Tree	0.281035	0.274091	0.203795	0.61963
Random Forest	**0.41048**	**0.509099**	**0.321518**	**0.63190**
MLP	**0.338126**	**0.531916**	**0.293429**	**0.66257**

Table 1: Performance of classifiers with MFCC features

Classifier	AP-P001	AP-P002	AP-P003	Accuracy
SVM	0.156369	0.213538	0.13358	**0.50920**
KNN	0.175214	**0.23592**	0.223585	0.33128
AdaBoost	**0.235867**	0.223081	0.22864	0.49693
Gradient Boosting	**0.2441**	**0.246032**	**0.282572**	0.48466
Gaussian Naive Bayes	0.174072	0.174072	0.211515	0.44785
Decision Tree	0.190729	0.177895	0.219198	0.49693
Random Forest	0.217545	0.217545	**0.273405**	**0.54601**
MLP	**0.352675**	**0.338022**	**0.356247**	**0.61963**

Table 2: Performance of classifiers with ASR features

- CPU time taken to create feature vectors : 50mins
- CPU time taken to train classifiers : Almost instantaneous
- AWS credits not used

Conclusion

The best accuracy of **66.257%** was obtained by training an MLP classifier on the MFCC feature vectors. I trained it like a Multi-class classifier, and this showed better results than training on 3 separate binary classifiers because in the latter, the values predicted by the 3 separate classifiers could lie in different ranges and scales. A multi-class classifier overcomes this shortcoming by predicting the scores of the 3 classes on the same scale. Hence, they proved to work better. Among all the classifiers, MLP, Random Forest and Gradient Boosting turned out to give the best results.

Future Work

Learning sound representations by capitalizing on large amounts of unlabeled sound using SoundNet.