11775-SG: Homework 1

YongKyung Oh*
Language Technologies Institute
Carnegie Mellon University
Pittsburgh, PA 15213
yongkyuo@andrew.cmu.edu

Abstract

The task of homework 1 is to perform multimedia event detection (MED) with audio features. Main tasks are extract MFCC and ASRs features from video file and develop model for multiple events. For the MFCC, Kmeans clustering is used to define features. In the ASRs, customized vocabulary dictionary using NLTK is used for features. SVM classifier is developed for the baseline and customized classification approach is suggested.

1 Introduction

- The overview of MED pipeline is depicted in Figure 1. In the first step, we extract features from the raw training data. In this homework, the audio features we will use are MFCCs and ASR transcripts.
- Implement the bag-of-words representation with k-means clustering. MFCCs features are extracted by openSmile. For MFCCs, probability of each K-means cluster is used as MFCCs features. Custom vocabulary is built by NLTK and bag-of-words representation is also used as ASR features.
- Support Vector Machine (SVM) is used to classify the video samples. Chi2-kernel is used. Additional to that, SMOTE oversampling is used to deal with imbalanced issue.

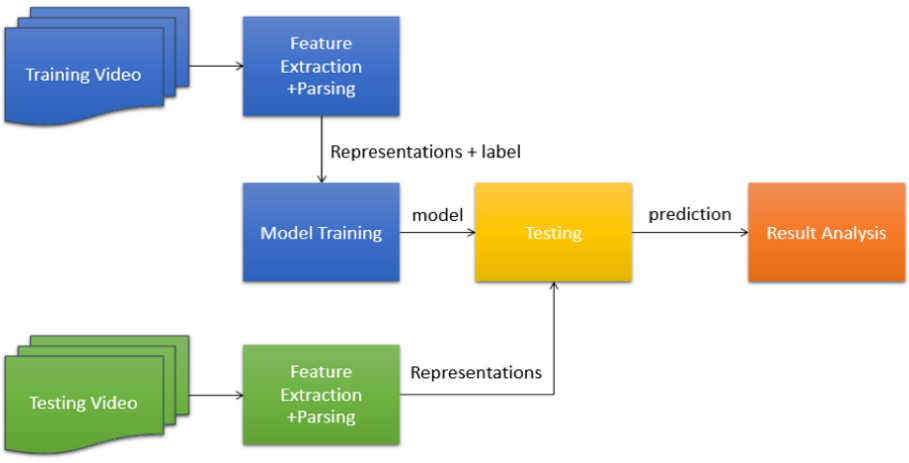

Figure 1: Project Pipeline

*UNIST, ok19925@unist.ac.kr

2 Data

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).

- For training, the file **all_trn.lst** specifies 836 training videos and their labels.
- For validation, the file **all_val.lst** contains 400 videos and their ground-truth labels as well. Validation set is used to tune hyper-parameters.
- For testing, there are additional 1699 videos specified in the **all_test_fake.lst**, in which their labels are all fake (deliberately set as NULL)

2.1 MFCC Features

Mel-frequency cepstrum (MFC) is a representation of the short-term power spectrum of a sound, based on a linear cosine transform of a log power spectrum on a nonlinear mel-scale of frequency. Mel-frequency cepstral coefficients (MFCCs) are coefficients that collectively make up an MFC.

2.2 Bag-of-Words Representation

Representing a video using bag-of-(audio-)words is one of the feasible approaches. To speed up the clustering process, you can choose only a small portion of the MFCC vectors (ex. randomly select 20% MFCCs from each video). To represent the video, K-means clustering is implemented. To determine the number of K, sum of squared error and silhouette score are used.

2.3 ASR Transcriptions

ASR transcriptions are provided, which include the text characteristics of the video. For the bag-of-words representation, txt format is only considered. Customized vocabulary is extracted by NLTK tokenizer. 8,006 features dimension is uses.

3 Experiments

3.1 K-means clustering

MFCC feature is captured by each frame with 39 different features. To speed up, 20% random sample dataset (select.mfcc.csv) is used to test and develop model. This train data has dimension (1999632, 39). Most of all, K-means clustering is implemented. To find the optimal K, two measures are used.[2]

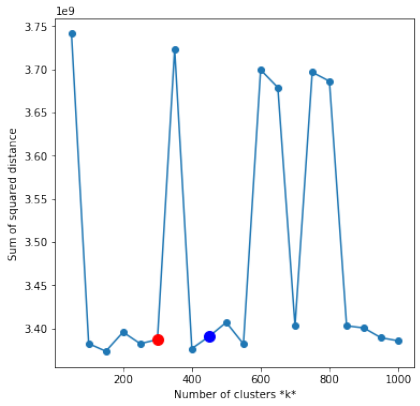

Figure 2: K means: SSE 50-1000

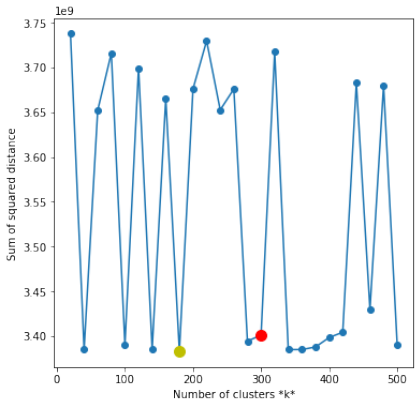

Figure 3: K means: SSE 20-500

[2]Implementation reference link for K-means clustering

First measure is sum of squared distance that is used to conduct elbow method. In the Figure 2, error is minimum at k=150. Also, the small region in the Figure 3, error is minimum at k=180. However, overall relation is inconsistent, so it is hard to use independently.

> Calculate the Within-Cluster-Sum of Squared Errors (WSS) for different values of k, and choose the k for which WSS becomes first starts to diminish. In the plot of WSS-versus-k, this is visible as an elbow. [1]

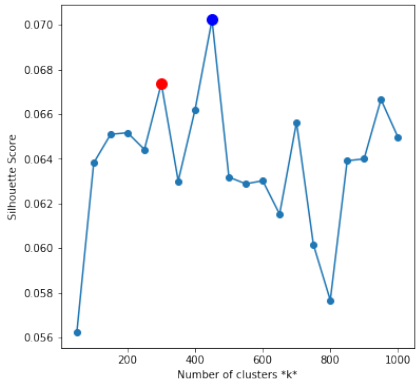

Figure 4: K means: SIL 50-1000

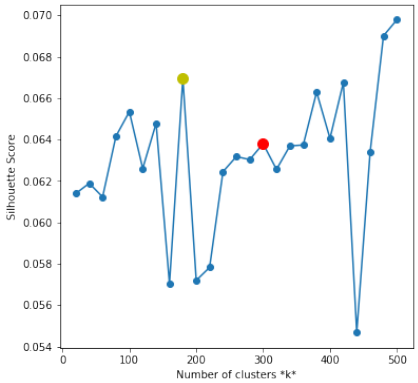

Figure 5: K means: SIL 20-500

Second, Silhouette score is calculated with 50,000 samples. Because silhouette score compare the intra-cluster distance and extra-cluster distance among samples, it require a lot of computation. To investigate the trend, relatively small number of samples are used. In the Figure 4, maximum score is at k=450. Also, the small region in the Figure 5, maximum score is at k=500.

> Silhouette refers to a method of interpretation and validation of consistency within clusters of data. The technique provides a succinct graphical representation of how well each object has been classified. [2]

Using this two measure, I tried to find the k around 150 to 300. In the final step, k is fixed to 180. There are two reasons. First, k means clustering contain randomness. I fixed the random state for reproducing, but the results are not consistent. Second, 180 features for SVM work better than other cases. So, I used 180 dimension features as a bag-of-representation for classifier.

Additional to that, current development environment (EC2 t2.large) is not suitable for the large computation and not available for GPU computation. To reduce the computational burden, I use the mini-batch k means with batch size 50.

> The MiniBatchKMeans is a variant of the KMeans algorithm which uses mini-batches to reduce the computation time, while still attempting to optimise the same objective function. Mini-batches are subsets of the input data, randomly sampled in each training iteration. These mini-batches drastically reduce the amount of computation required to converge to a local solution. [3]

MiniBatchKmeans has little bit worse result than original Kmeans. Kmeans cannot handle larger dataset. For example, if I try the cluster number 180 using Kmeans, then memory error occurs. Instead of the accuracy (or precision) performance, I use the larger feature dimension to develop the better classifiers.

3.2 Support Vector Machine (SVM) classifier

Before develop classifier model, I conducted re-sampling pre-process to deal with imbalanced issue. For example, list of train data contain 836 video name and label. 816 out of 836 data has MFCC features and only 27 data labeled as event 1 (P001). So, the imbalanced ratio is $0.0342 (= 27/789)$. In this binary classification case, model should developed by minor class features. Class weight and

re-sampling technique can be considered. In this case, SMOTE oversampling is used to sample the minor dataset.

> SMOTE: Synthetic Minority Over-sampling Technique. To illustrate how this technique works consider some training data which has s samples, and f features in the feature space of the data. Note that these features, for simplicity, are continuous.
>
> To then oversample, take a sample from the dataset, and consider its k nearest neighbors (in feature space). To create a synthetic data point, take the vector between one of those k neighbors, and the current data point. Multiply this vector by a random number x which lies between 0, and 1. Add this to the current data point to create the new, synthetic data point. [4]

For the classifier, support vector machine (SVM) classifer is used. SVM is a discriminatory classification formally defined by the separation hyperplane. In other words, when labeled train data (supervised learning) are given, the algorithm outputs an optimal hyperplane to classify new data. To tune the optimal hyper parameters, following factors are considered. [3].

- Kernel: Specifies the kernel type to be used in the algorithm. Chi2-kernel is selected, because the chi2-kernel do some normalization in the kernel itself, so it's often better than other kernels, especially for the such a large scale data.
- C: Regularization parameter. The strength of the regularization is inversely proportional to C. Instead of default 1, 2.0 is determined by experiments. More penalty to find better prediction for minor set.
- Class weight: Set the parameter C of class i to *class_weight*$[i] * C$ for SVC. Balanced class weight is used. Sampling is conducted, so it may not considered as well.

3.3 Light GBM classifier

> LightGBM is an open-source framework for gradient boosted machines. By default LightGBM will train a Gradient Boosted Decision Tree (GBDT), but it also supports random forests, Dropouts meet Multiple Additive Regression Trees (DART), and Gradient Based One-Side Sampling (Goss). [5]
>
> Boosting train models sequentially that each model learns from the errors of the previous model. Starting with a weak base model, the model is repeatedly trained, each in addition to the previous model's predictions, producing a strong overall prediction. [4]

Light GBM is widely used to deal with larger data with faster speed and performance. Grid search cross validation is implemented. According to the multiple experiments, different parameter classifiers works well in the different event settings. Therefore, I conducted multiple grid search CV for each event. I consider following parameters: num_leaves, $min\_data\_in\_leaf$, $lambda\_l1$, and $lambda\_l2$.

4 Results

SVM model, pre-tuned Light GBM and parameter tuned Light GBM are used to compare the results. Parameter tuned Light GBM tend to overfit into train data. Therefore, the performance is worse than that I expected. Also, there are 3 cases of using features: MFCC only, ASR only and all features (both MFCC and ASR). The best cases are as follow using validation set (*all_val.lst*):

- Event 1 (P001): SVM model with all features (both MFCC and ASR)
- Event 2 (P002): SVM model with MFCC features
- Event 3 (P003): LGBM model with all features (both MFCC and ASR)

[3] Implementation reference link for SVM

[4] Implementation reference link for Light GBM

	MED	MFCC	ASR	ALL
SVM	P001	0.068333	0.048333	**0.101667**
	P002	**0.348232**	0.045000	0.151111
	P003	0.105901	0.076739	0.139457
LGBM	P001	0.051667	0.037500	0.057222
	P002	0.280000	0.045000	0.204167
	P003	0.076283	0.158357	**0.276715**
LGBM (Tuned)	P001	0.051667	0.037500	0.101667
	P002	0.329969	0.045000	0.120833
	P003	0.115217	0.146848	0.269384

Table 1: Result of MAP(Mean Average Precision)

As expected, different model works well in the different events. In the case of MFCC, feature dimension is not enough to improve the classifier performance. It is very hard to increase the k with current development environment, different approach may be required. Also, neural network approach may work better in this setting, but I couldn't conduct further experiments due to memory issues.

Final score file is produced by using test set (*all_test_fake.lst*). SVM and Light GBM model is used to generate score file. Total 18 files (3 data * 3 event * 2 model) are generated. Best file is selected by the result of validation set.

References

[1] Thomas J Archdeacon. *Correlation and regression analysis: a historian's guide*. Univ of Wisconsin Press, 1994.

[2] Sam T Roweis and Lawrence K Saul. Nonlinear dimensionality reduction by locally linear embedding. *science*, 290(5500):2323–2326, 2000.

[3] David Sculley. Web-scale k-means clustering. In *Proceedings of the 19th international conference on World wide web*, pages 1177–1178, 2010.

[4] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002.

[5] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. In *Advances in neural information processing systems*, pages 3146–3154, 2017.