Semantic Concept

Supervised framework for Semantic concept detection

- For "noiseme" annotation from CMU, refer to the paper "Exploring Audio Semantic Concepts for Event-based Video Retrieval" submitted to 2014 ICASSP

- Extend the framework on annotations offered by other teams

We cleaned the code for training audio semantic concept classifier and apply the pipeline to the annotation provided by other teams. We only make use of the annotation on MED dataset, including SRI-Sarnoff's annotation (320 clips, 28 concepts) and SRI's annotation (671 clips, 20 concepts). We exclude rarely appeared concepts from the annotation. We use 2s window for acoustic feature extraction. We apply the same feature generation and late fusion strategies (late fusion of segment 1,3,5,10; each segment: mean and variance) in noiseme detection pipeline on each annotation. The result for each annotation is listed in the table below.

	MAP	Pmiss@TER=12.5	minNDC
SRI-Sarnoff 320clips-28 concepts (Use 25 in detection)	0.1548	0.6424	0.8739
SRI 671 clips-20 concepts concepts (Use 19 in detection)	0.1283	0.6454	0.9037
Noiseme 2s (Use 40 in detection)	0.1423	0.6477	0.8944
Noiseme fusion of 2s and 0.5s (Use 40 in detection)	0.1517	0.6308	0.8855

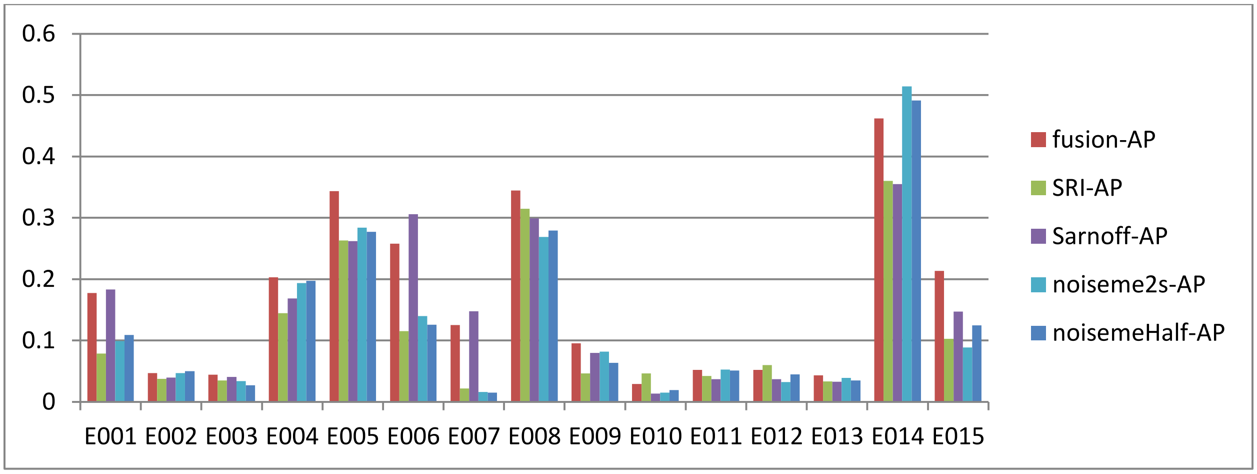
We can observe that the pipeline using SRI-Sarnoff's annotation and the noiseme annotation outperform using SRI's annotation in terms of MAP. But all of them achieve similar performance in terms of Pmiss value.

We take the late fusion of the system using each annotation and the result is shown in the table below. We can find out that each annotation provide complementary information in distinguish different event and thus the fusion experiment get improvement.

	MAP	Pmiss@TER=12.5	minNDC
Fusion Sarnoff25, SRI19	0.1694	0.6166	0.8624
Fusion Sarnoff25, noiseme2s, noiseme0.5s	0.1740	0.6118	0.8595
Fusion SRI19, noiseme2s, noiseme 0.5s	0.1598	0.6169	0.8740
Fusion Sarnoff25, SRI19,	0.1787	0.5961	0.8555

noiseme2s, noiseme0.5s			

We also compared the performance for each event. The figure below indicates that the performance is mainly influenced by the quality of the concept instead of quantity. Quality refers to how much the concept is related to the semantics in the video or how effective it is to distinguish this event from others. For example, we have 40 concepts in noiseme annotation but we didn't achieve significant improvement than Sarnoff's annotation, which including 25 concepts. Also, the SRI annotation with only 19 concepts outperforms other system for some events. We will further investigate effective strategies to combine different annotation.



Analysis of Semantic Concept Effectiveness for event classification

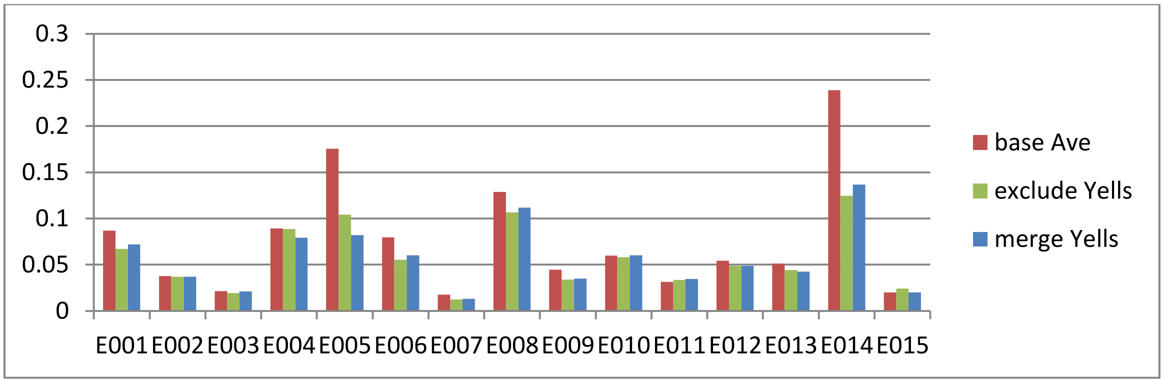
We used multiple annotations of audio semantic concepts in previous experiment and got improvement by fusing system with different annotations. But how to define effective semantic concept is still an unsolved question. Therefore, we further investigated the influence of some concepts based on the analysis of the characteristic of the each annotation and observe how they influence the performance.

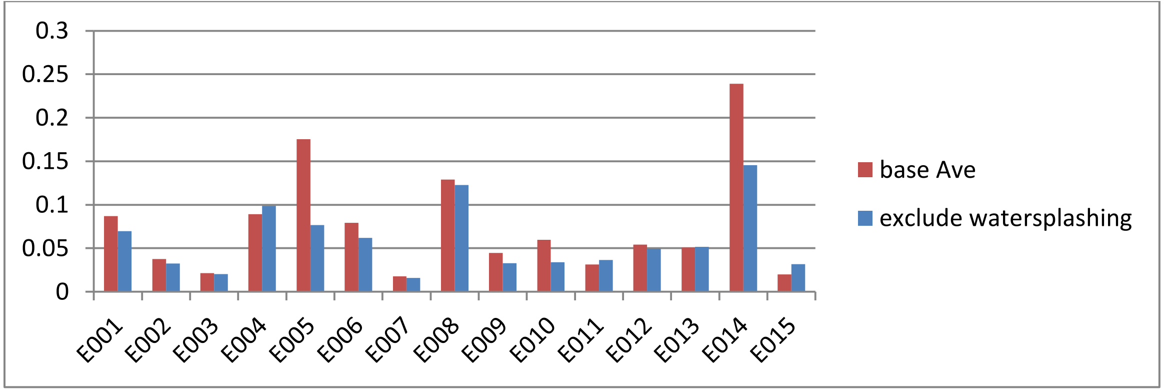
The method follows the principle of feature selection. We either exclude the semantic concept from the feature or merge several into one to examine the performance. The more degradation of the performance from the baseline system, the more effective the concept or the splitting of the category is for distinguish semantics in clip level. (The baseline performance with all concepts is highlight in each table)

For annotation offered by SRI with 20 concepts, the overall performance for each experiment is listed in the table below. This annotation has multiple Individual and crowd pairs of certain concept, e.g. "Individual Laugter" and "Crowd Laughter". To examine its influence, we merge them by taking the average of the confidence for each concept. The results show that this definition is meaningful for MED task since there is significant degradation. The concept of "Yells" and "WaterSplashing" doesn't appear in noiseme annotation so we also test their effectiveness.

Med11	MAP	Pmiss
SRI Ave	**0.0818**	**0.7033**
merge Laughter	0.0737	0.7089
merge Running	0.0734	0.7097
merge Cheers	0.0698	0.7177
merge Applause	0.0644	0.7216
exclude Yells	0.0638	0.7411
Exclude WaterSplashing	0.0634	0.7337
merge Yells	0.0629	0.7414

We further compared the performance for experiment with significant change for each event in terms of AP. The figures below show that both “Yells” and “WaterSplashing” are more useful for the detection of E005(working on a wood project) and E014(repairing an appliance).



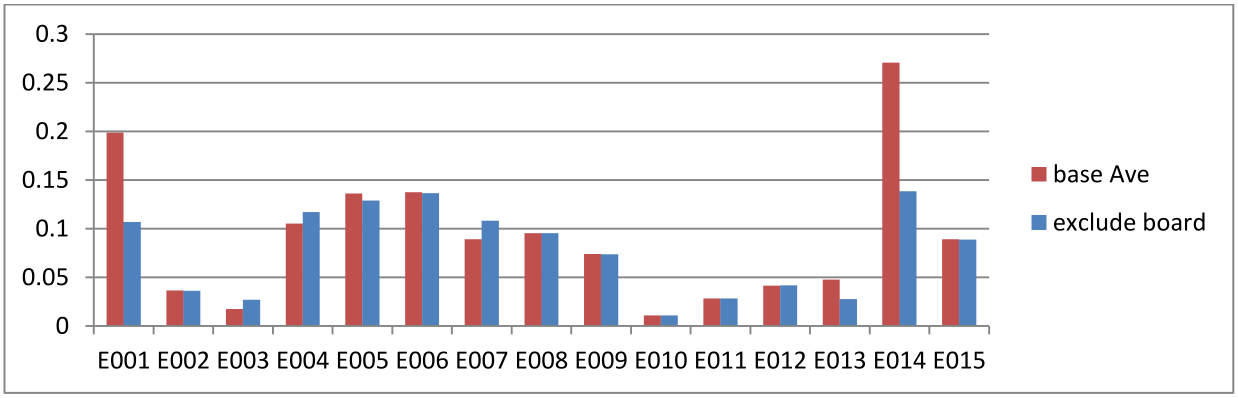


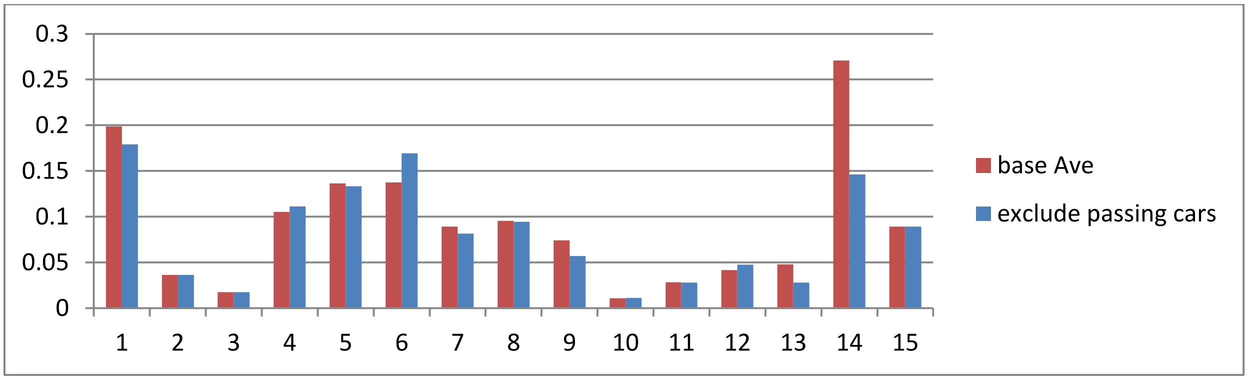
For annotation provided by SRI-Sarnoff with 28 concepts, we merge different types of speech (instructional, conversational, someone gives a speech). We also examined several concepts special for this annotation. From the table below, we can see that several concepts don’t contribute much to the performance. Only the distinguish of different speech, the concept “noise_of_passing_cars” and “board_hitting surface” bring significant change to the performance.

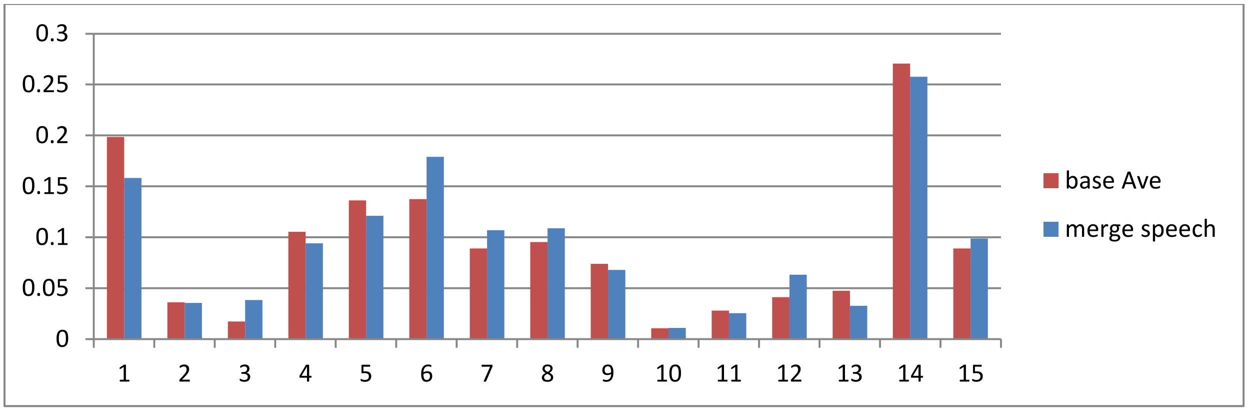
Med11	MAP	Pmiss

Sarnoff Ave	**0.1047**	**0.6966**
exclude marching_band	0.1044	0.6962
exclude group_dancing	0.1023	0.6975
exclude word_tire_spoken	0.1023	0.6861
exclude word_how_to_spoken	0.0995	0.7002
merge speech	0.099	0.6837
exclude noise_of_passing_cars	0.0961	0.6964
exclude board_hitting surface	0.0885	0.7051

Further investigation of the performance for each event, we find out that “board_hitting_surface” mainly influences E001(attempting a board trick), E013(parkour), E014(repairing an appliance). The concept “noise_of_passing_cars” mainly influences E013(parkour) and E014(repairing an appliance).





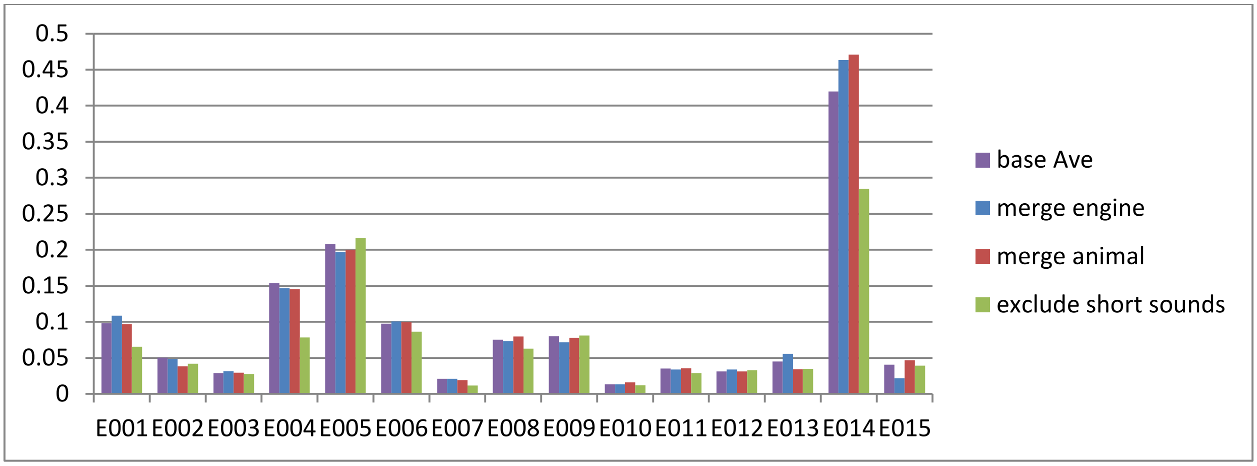


The interesting fact is that removing the concept or merging several directly might even bring some improvement for event detection. For example, merging speech makes the detection of E003, E006, E007, E008, E012 better. This might be due to some acoustic similar sound with different semantic meaning make the concept detection more difficult with insufficient training data. The unreliable detection result will bring into only noise that affects the performance.

For the noiseme annotation with over 40 concepts, we investigated merging different engine sound, animal sound and multiple sounds which appear in a short time, e.g.click, hammer, knock, etc. The result shown in the table below indicates that the sub-categories of animal and engine don’t contribute to the event detection system. Although the sub-categories are reasonable defined with different acoustics, the rare examples make it hard to train classifier and get reliable detection result.

There are over 10 concepts belong to the short sound class and ignoring all of them will affect the performance. So this means that merging some of these concepts with similar acoustic features might benefit the system since each merged concept will have more samples.

Med11	MAP	Pmiss
merge animal	0.1047	0.6723
merge engine	0.1013	0.6757
Noiseme Ave	**0.1012**	**0.6768**
merge short sounds	0.0975	0.6997
exclude short sounds	0.0885	0.7051



Expand Semantic Concept Vocabulary

Semi-automatic framework

Refer to the paper “Semi-automatic Audio Semantic Concept Discovery for Multimedia Retrieval”, submitted to ICASSP 2014

Repeated sequence discovery

In previous framework, people often define a certain number of sound events (audio semantics) and classified the sound into predefined categories. However, there often appear some out-of-vocabulary sounds in new events. Also, labeling enough data for supervised training would be time consuming and expensive.

If these sound events are representative of the event, they would occur multiple times in different videos of the event. Therefore, studying unsupervised method to discover such repeated pattern automatically is meaningful for semantic vocabulary enlargement as well as reducing the cost of labeling. An unsupervised pattern discovery approach [1] is proposed in speech processing for unsupervised word acquisition. It utilizes a segmental variant of dynamic time warping technology (segmental DTW algorithm) to find matching acoustic patterns between spoken utterances. We tried to apply this method in sound event acquisition.

In order to find whether this idea can work on our data, we did some investigation. We extracted the audio segments of certain noisemes, like hammar, laugh, tone, etc. Each segment is represented by a sequence of MFCC features. The MFCC feature is the decoding result from our 4096 codebook instead of the raw feature. We precomputed the distance between different centroids of MFCC clusters. Then, We calculated the distance matrix between audio segments. In the distance matrix, each cell corresponds to th Euclidean distance between frames from each of the utterance being compared. If they contain some similar segments, we should find some diagonal warp path in the matrix.

From Figure 1, which shows the distance matrix between “hammer” segments, we can see that the distance matrix is evident in some pattern structure. Listening to the segments, we know that there is 3 and 5 separate “hit” sound in each segment, which is evident in the figure. But we cannot observe obvious diagonal lines as we expected, though some points seem to construct distorted fuzzy line.

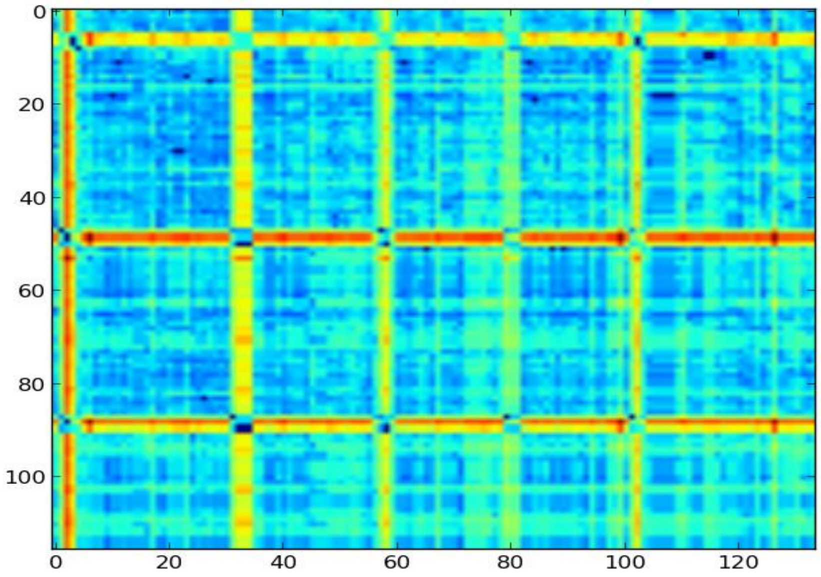


Fig1. Hammer segment from HVC013188, 6.949 8.100 ; 54.629 55.969

As is shown in Fig2, the self similarity distance matrix of one segment we used above give us some encouraging result. There exist some diagonal lines clearly, which means the corresponding sub segment can be recognized as similar segments in this method.

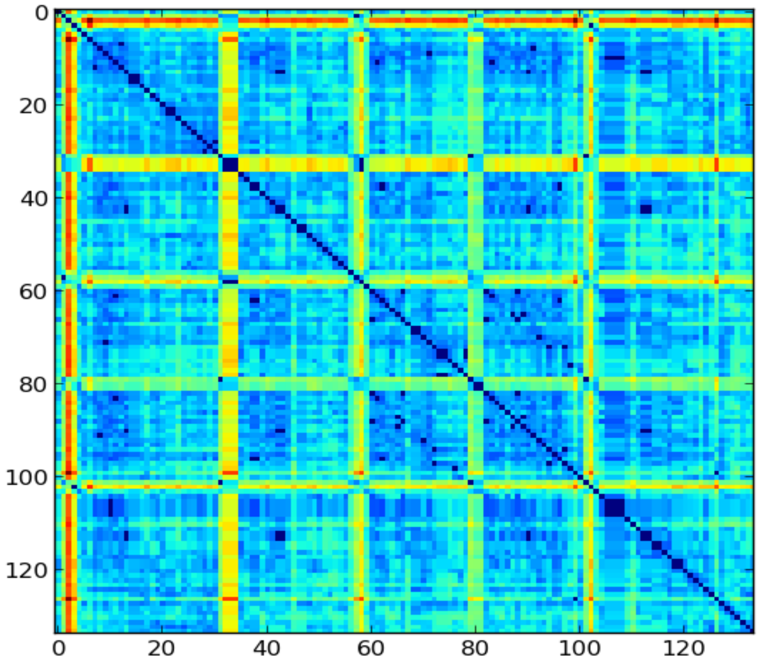
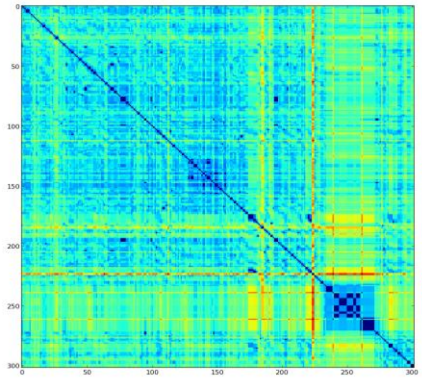
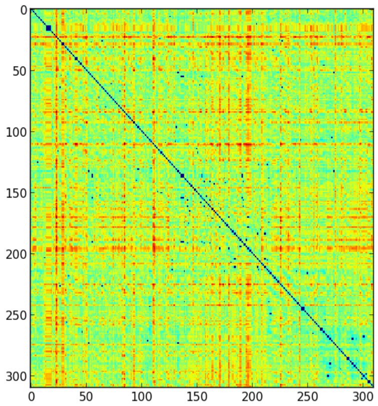


Fig 2. Hammer segment from HVC013188, 6.949 8.100 ; self-similarity

We also compare other segments from different noisemes. The result is in the chart below. Although human can distinguish the similar repeated segments clearly, the distance matrix cannot reveal this. The main reason is that the background noise disturb the acoustic features. Also, the variance of the sound affect the result. This phenomenon is like the same word spoken by different speakers or from same speaker but in different prosody.

	Laugh HVC014770 54.050 57.076

	Scream HVC022063 17.018 20.116

[1] Alex S. Park and James R. Glass, Unsupervised Pattern Discovery in Speech, IEEE transactions on Audio, Speech,and Language Processing, VOL. 16, NO.1, January 2008