Taming the Wild: Acoustic Segmentation in Consumer-Produced Videos

Benjamin Elizalde and Gerald Friedland

TR-12-016

January 2013

Abstract

Audio segmentation is the process of partitioning data and identifying boundaries between different sounds, this task is commonly an early stage in speech processing tasks such as Automatic Speech Recognition (ASR) or Speaker Identification (SID). While traditional speech/non-speech segmentation systems have been designed for specific data conditions such as broadcast news or meetings, the growth of web videos brings new challenges for segmenting consumer-produced, aka ``wild," audio. This type of audio is an unstructured domain with little control over recording conditions. Despite the growth of ``wild" audio, little research has been done on this domain or on domain-independent audio segmentation systems. The following paper attempts to close that gap by creating and testing a semi-supervised approach with a Codebook-Histogram Features (CHF) segmentation using Support Vector Machines (SVM) for speech detection in consumer-produced videos. Using the web videos TRECVID MED 2011 dataset and a well-known speech detection meetings corpus, training/testing data combinations were designed to evaluate and understand better the performance of this new approach in contrast to a state-of-the-art traditional Gaussian Mixture Models (GMM) system. The results revealed that the CHF approach outperformed the GMM method by 50% detecting speech on meetings, but underperformed it by 44% on wild data. Furthermore, the CHF was 4 times faster at processing audio files at the testing stage.

International Computer Science Institute, 1947 Center Street, Suite 600, Berkeley, California, 94704

1. INTRODUCTION

Audio segmentation is a fundamental task in almost all fields of speech processing such as ASR, SID, and others such as Speaker Detection aka Diarization. It consists in drawing accurate boundaries between segments, which helps to filter relevant information and making them more meaningful for systems to process. The classic application for a segmentation task is what we know as Speech Activity Detection (SAD), also known as Voice Activity Detection (VAD). It essentially involves the detection and labeling of speech and non-speech segments in a given audio file.

Segmentation in controlled and single domain data is a developed field, yet it's still an unsolved task when dealing with data that doesn't adhere to specific rules or structures such as the audio from web videos. With the Internet more widespread and mobile gadget recording capabilities, the uploading of this type of videos is increasing. Let's take, for instance, one of the most popular websites for consumer-produced videos, YouTube, which claims that 72 hours of video are uploaded every minute, resulting in nearly 8 years of content uploaded every day,[1] and this is only one of many examples. Consumer-produced media mostly contain low quality recordings, noisy environments, overlapping sounds, a variety of languages, and no specific audio structure. However, if one wants to filter the relevant information for systems to process, traditional segmentation approaches might not be able to address them.

High accuracies have been reached for traditional, corpus-based, supervised segmentation tasks. Nevertheless, in wild audio one cannot rely on any single characteristic to draw boundaries between classes and it is difficult to pre-train models because of the high variance in the data. Supervised techniques should have enough data to try to compensate this variability, yet there aren't enough well annotated databases of this type. On the other hand, unsupervised approaches deal better with these problems because they use little to no training data. These techniques learn from the testing data only, therefore they don't gather as much data to provide the same level of accuracy as their counterpart. Traditional approaches take advantage of the statistical models, such as the Gaussian Mixture Models (GMM) and Hidden Markov Models (HMM). These models provide a robust solution, but are normally iterative and require important computational power. When it comes to large-scale data it is important to speed up the processing time.

In this paper we present a hybrid segmentation system with both an unsupervised and a supervised perspective– a codebook and sets of labeled speech/non-speech segments respectively. The labeled segments lack the sufficient variation existing in the consumer-produced audio, but provide a learning step for the algorithm. Thus, the objective is that the codebook provides a background structure to the labeled segments, compensating the diverse characteristics of the wild audio. The combination output are the new CHF, which are also fast to process because they are one-dimensional in contrast to the typical multidimensional Mel Frequency Cepstral Coefficients (MFCC), which tend to have from 13 to 58 dimensions. The classification of the CHF was carried by an SVM, which determines if the segment, represented by a CHF, is speech or non-speech.

In order to compare the performance of the new approach, a state-of-the-art traditional GMM based system was considered. We used the TRECVID MED 2012 to test both approaches designing training/testing data combinations to aid the understanding of what speech/non-speech detection on consumer-produced media entails and how these traditional approaches to audio detection performed in this wild domain. We also included the most relevant experiments that served to tune the CHF system.

We structure the article as follows. Section 2 commences presenting some related work. Section 3 presents an overview of the system, while Section 4 describes the nature of the data and the experimental setup. Section **??** presents the tuning experiments of the CHF System. Section 6 submits the results and the corresponding analysis before Section 7 resumes with the conclusion and the outlook for the future.

2. RELATED WORK

Speech/non-speech segmentation has been well studied in controlled domains such as meetings and broadcast news, which involve the presence of background music and occasional noise, as summarized in.[2] Most of the model systems are based on HMMs modeled by GMMs.[3] Some applications of GMMs are also found on very diverse sets of audio features[4] but in spite of that, SVMs remain rarely used for this task despite their good discrimination efficiency.[5] Originally, SVMs impose a constraint of two-class discrimination, but various solutions have been proposed to extend the SVM into a multiclass classification method.[6]

A more complex data domain currently under investigation is used in the DARPA RATS program.[7] This data is collected under both controlled and uncontrolled field conditions over highly degraded, weak and/or noisy communication channels. For this scenario,[8] analyzes the performance of combining a one-layer NN and a GMM-based system along with an emphasis on feature extraction, which includes long span features and acoustic PLP features.

For consumer-produced data, research related to content analysis tasks includes speech activity detection, systems which use traditional hierarchical GMM based-algorithms. An example of the SAD task employing web videos[9] compared a GMM-based system using Mel Frequency Cepstral Coefficients (MFCCs) against a Maximum Entropy (MaxEnt) system using an alternative set of spectral and energy features. The article concluded that the MaxEnt and the set of alternative features yielded a lower error.

Despite this early promising conclusion that using a non-traditional method could bring better results, there seems to be no work that systematically evaluates different speech/non-speech classification algorithms on consumer-produced videos. We therefore concluded to present a new approach in this article.

3. DESCRIPTON OF THE SPEECH/NON-SPEECH DETECTION SYSTEMS

The following section describes the state-of-the-art GMM-based system and the CHF-based approach.

3.1 GMM-based system

In order to analyze the effectiveness of GMM systems on consumer-produced audio, we utilized the SHOUT[10] speech/non-speech system described in Figure 1, which is considered state-of-the-art for meeting recognition.[11] It has two main characteristics. First, the system learns from the testing set instead of the training set, which only serves to train generic speech and non-speech models. The "out of the box" version of the system trained these models with more than 30 hours of Broadcast News data and it is not necessary to retrain them with other data. Second, this system does not have parameters to tune. This combination of characteristics makes SHOUT a self-sufficient, almost unsupervised algorithm and therefore an excellent option when little-to-no training data is available. During the training stage, two bootstrap GMMs must be first trained, one for speech and one for non-speech. In the testing stage, after the feature extraction of the audio, a bootstrap segmentation of speech and non-speech is performed using the training models. This segmentation is used to train silence and audible non-speech GMM models iteratively from the non-speech segments. After, a speech model is similarly trained from the speech segments. Once the three models are obtained, the necessity of the audible non-speech model is reviewed. This check is performed using the BIC, which is a penalized version of the maximum likelihood approach. The speech model and the audible non-speech model are then compared to see if they are the same. If they are, then the audible non-speech model is not needed and it is discarded while two new models are created speech and silence, if not then all of them are kept. This is performed iteratively to either merge the models or discard them. After merging the models, a retraining of the GMMs is performed. The segmentation decision is performed using the Speech Model only.

3.2 CHF-based system

The system[12] is divided in two stages of training and testing. For the training stage, there are two subsections. The first is an unsupervised section where a codebook is created with the output of a compilation of MFCCs that are fed into a K-means algorithm. The second is a supervised section in which the audio and the ground truth are used to create one set of labeled MFCC segments for speech and one set of labeled non-speech. The codebook and the sets of the labeled segments are then related to create histograms, by determining the closest

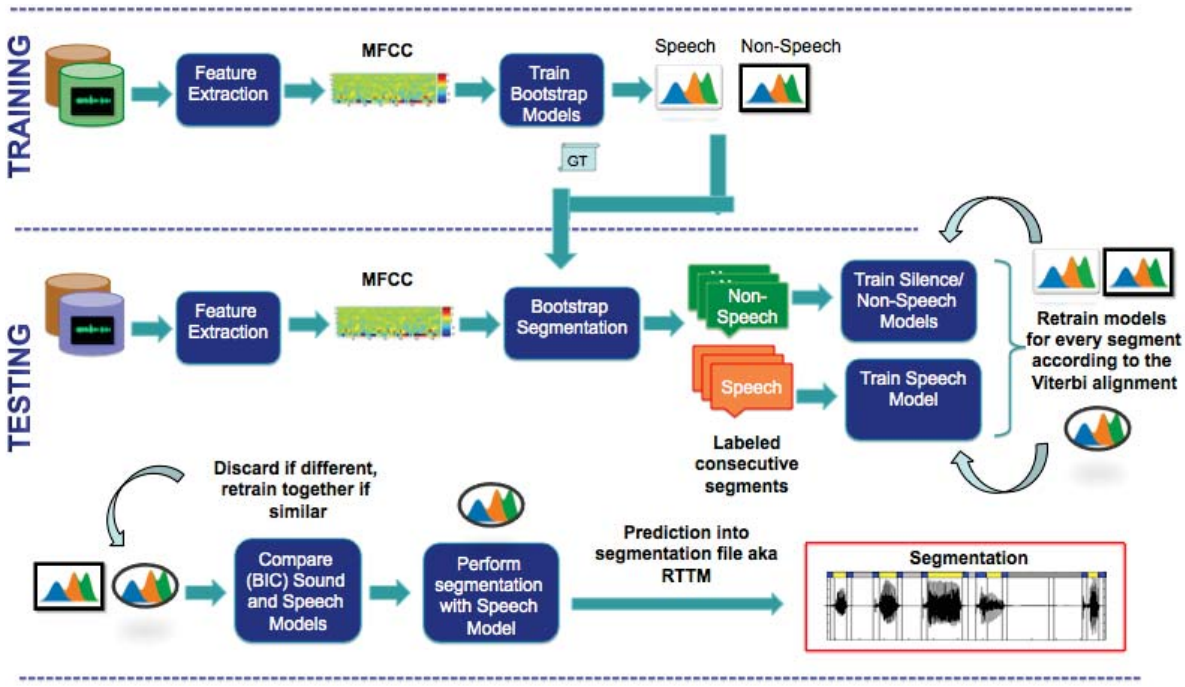

Figure 1. SHOUT speech activity detection algorithm.

codebook cluster per frame for each segment. The number of histograms sets is determined by the number of sets of labeled segments–one set for speech and one for non-speech. These histogram sets are used to train a SVM model using a radial basis function kernel. For the testing stage, a set of unlabeled consecutive test segments for each audio file is created, which later is transformed into histograms in the same manner as the training stage. These histograms are classified by the SVM using the previously mentioned SVM model. The output scores given by the SVM are then used to create the segmentation files.

4. DATA DESCRIPTION & EXPERIMENTAL SETUP

For the analysis of the systems herein, we chose to compare the performance of the two above described systems using a meeting dataset against the TRECVID 2012 consumer-produced video dataset. The datasets are described as follows.

4.1 Data

The ICSI Meeting Corpus is a collection of 74 meetings including simultaneous multi-channel audio recordings, word-level orthographic transcriptions, and supporting documentation collected at the International Computer Science Institute in Berkeley, ICSI. The meetings included are "natural" meetings in the sense that they would have occurred anyway. The meetings included here generally run just under an hour. This type of audio has little to no background noise or any other type of audible non-speech. The dataset is widely known as a benchmark for diverse speech tasks and has been used in numerous NIST Rich Transcription evaluations.[13]

On the consumer-produced audio side, we used a subset of the NIST TRECVID MED 2012 video database called DEV-T. The entire dataset comprises a collection of training and testing data for a total of 150,000 video files of about three minutes each and with only 14 hours of annotated data. It is organized around 30 concept classes such as "Board Trick," "Feeding an Animal" or "Landing a Fish." In this type of data there is not a fixed structure. Music, unstructured speech, far field speech, high level background noise, and other examples of challenging to segment audible sounds may be encountered with some hard to identify and even unknown by human annotators.

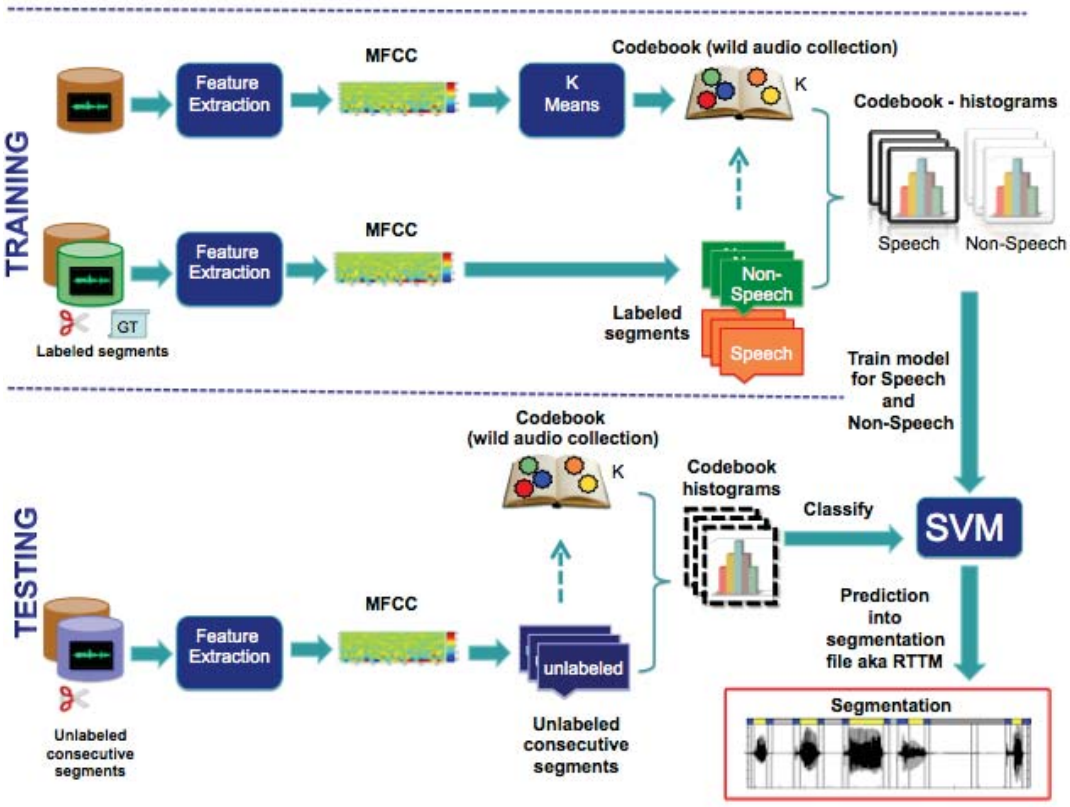

Figure 2. CHF speech activity detection algorithm.

A selected 12 hours subset from the TRECVID's DEV-T subsection and from the Meeting corpora that included annotations for speech regions were used for the experiments. In this article, meeting recordings will be referred to as "clean" data and the TRECVID 2012 as "wild" due to their above described characteristics. All audio had a sample rate of a 16kHz, 1 channel, PCM format. The extracted audio features were typical 19 Mel Frequency Cepstral Coefficients plus Delta for a total of 38 dimensions, with a 25 ms window and a window step of 10 ms.

4.2 Experimental Setup and Error Metric

In order to normalize for the different characteristics of both tested systems and to get an idea of their cross-domain adaptability, our experiments consisted of four training/testing combinations sets, one for each system for a total of 8 runs. The first combination set was clean/clean and was planned to test the systems with the best-case scenario and obtain the performance baseline. The second and third sets included the combination of both datasets to observe how the systems behaved on mismatched conditions with consumer-data. The fourth and most important set, wild/wild, was intended to evaluate the best segmentation results in wild conditions. For each of the four experiments, a six-hour subset of each database was used for the training stage and a different six-hour subset for the testing stage.

In the fourth and last experiment two aspects of the processing time were compared–training and testing. Because the CHF have a simple parallelization capability allowing the user to process several files from end to end at the same time, which the SHOUT system does not, only one file at a time was processed for each of the systems at the both stages. Therefore, none of the parallelization benefits were employed during either stage. The data used for the training stage was the same for the fourth experiment and the testing data was a one-hour file from the meetings database. The three systems were run in a Dual Core AMD Opteron Processor

875 computer. The technical specs of this 64-bit machine included two CPU cores at a frequency of 2.2 GHz and 32 GB RAM.

The SAD systems were evaluated with the S_{error} % measurement.[13] This percentage relates the two speech error types: (MD) is the *Missed Detected* speech, or the total time of speech that was not classified as speech, and *False Alarm* (FA), which is the total time of non-speech that was falsely classified as speech. Lastly, S_{total} is the total time of speech in the ground truth. The equation writes as:

$$SAD_{error}\% = (MD + FA)/S_{total} * 100 \quad (1)$$

While there are different distributions of speech and non-speech in the two datasets, this error metric converges at 45 % for the wild audio and 18 % for the clean audio, which is the expected random guess.

5. CHF SYSTEM'S TUNING

In the following section we executed several experiments to determine the best codebook version based on accuracy performance. Another set of experiments was performed in order to select the best SVM kernel and SVM parameters.

5.1 Choosing the Best Codebook Resolution

The codebook is a compilation of three hours of wild audio, extracted from the DEV-T database. For the computation of the codebook we used the *Matlab* K-means native function from the 2010 release. The experiment consisted in changing the initial K-cluster value from 16 to 1024. Each value had a different number of iterations that depended on the convergence, which was determined by the *Elbow Criterion*.[14] This criterion says that if you graph for each K-value the variance of a reference value per iteration, the first iterations will have a greater variance, but at some point the gain will drop, showing an angle in the graph, the "elbow". In our experiments, this elbow joint coincides with a decrement of about less than 20% in the reference value. The reference value used is the Euclidan distance between the data points and their cluster centroids.

The table 1 shows the different K-values tried for our codebook, it also includes the iteration that corresponded to the elbow joint, and the approximate computing time per iteration.

Number of clusters	Number of Iterations	Minutes per Iteration
1024	10	90
512	10	49
256	10	39
128	9	21
64	9	15
32	9	8
16	7	5

Table 1. Codebook iterations

Once the seven different codebook versions were created, a set of experiments was designed to analyze the performance of the system at classifying speech and non-speech segments. The experimental setup is similar to the first set of experiments mentioned in the Subsection 4.2. The only difference was that a Cross Validation set of similar size was used instead of the Test set. No SVM parameters were tuned and three different kernels were tested: Linear, Intersection, and the Radial Basis Function. The prediction values output from the SVM were compared against the ground truth to compute the accuracy of the speech segmentation. The results are shown in Figure 3 the Table 2.

The intuition behind K-means tells that more resolution will provide better results, but we needed to be sure that this was true for our algorithm. There was also a hypothesis suggesting that since we are classifying only two classes, only a few K-clusters would be sufficient to provide high accuracy levels. We've demonstrated here how more K-clusters and more resolution means higher accuracy values.

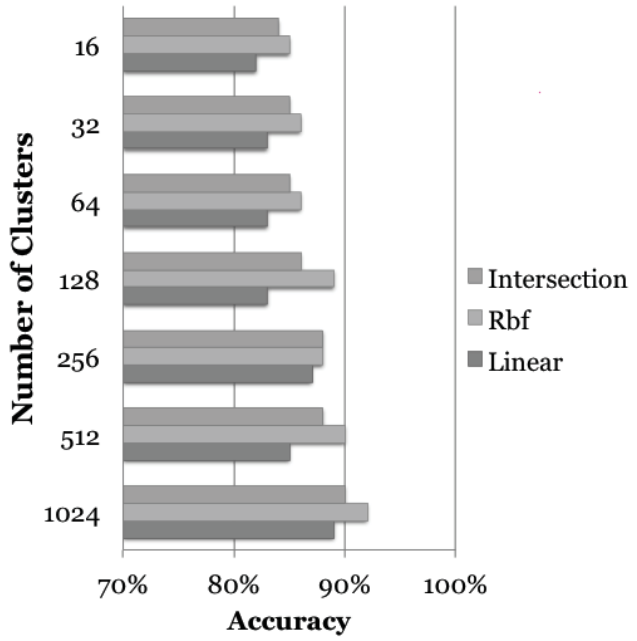

Figure 3. Accuracy values for different K-values and different kernels

Number of clusters	Linear %	RBF %	Intersection %
1024	89	92	90
512	85	90	88
256	87	88	88
128	83	89	86
64	83	86	85
32	83	86	85
16	82	85	84

Table 2. Accuracy values for different K-values and different kernels

More important to this conclusion, we have to observe the difference between the accuracy levels of using 1024 and 16 K-clusters. We can see how the difference is about a 6% improvement. This is not really a significant improvement considering that 1024 is about 64 times more K-clusters than 16.

Furthermore, in the table we can observe that the difference in time per iteration between 1024 and 16 is considerably higher. The accuracy values suggest that working with fewer K-clusters and including an optimum SVM tuning could end up in a descent and very fast segmentation. Our intention is to see the full potential of the algorithm, therefore the 1024 codebook version is used for the experiments in this paper.

5.2 Tuning the SVM

In order to tune the SVM we used the four set of experiments designed in Section 4 The kernels used to create the SVM Model were Intersection, RBF, Linear, and Chi-Square. In addition to this a Cross Validation tuning was done to compute the optimum parameters of "-c", which is the tradeoff between the training error and the margin and "-g", which defines how far the influence of a single training example reaches, with low values meaning 'far' and high values meaning 'close'. In these experiments we used Cross Validation sets for both of the data types similar to the Test sets described earlier. The results are shown in the Table 3.

The best values were achieved using the RBF kernel and the using a less complex kernel such as the Linear type, provided relatively closer classification values. Generally RBF will require more processing time than the

Set	Train/Test	RBF %	Linear %	Intersection %	Chi Square %
1st	Clean/Clean	7	9	20	21
2nd	Clean/Wild	37	38	38	39
3rd	Wild/Clean	11	13	19	21
4th	Wild/Wild	37	38	39	39

Table 3. SAD error results for the four experiments

Set	Train/Test	SHOUT %	CHF %
1st	Clean/Clean	15	7
2nd	Clean/Wild	21	36
3rd	Wild/Clean	20	10
4th	Wild/Wild	26	37

Table 4. Speech/Non-Speech Error results for the 8 experimental runs.

Linear counterpart, but since we wanted to test the system with the best tuning, we chose the RBF for the experiments presented in this paper. Further analysis on these results is discussed in the following Sections.

6. RESULTS & ANALYSIS

The results on the performance of the four sets of experiments are shown in Table 4. The results on the computing time for training and testing are shown in Figure 4. This figure presents the time between processing the input audio into a segmentation output for the testing stage. For the training stage, the processing time between the input set of audio and the corresponding training model for each system is considered. Note that the CHF system's training time does not include the codebook creation. Lastly the Figure 5 contains the results of the FA and MD errors for each set of experiments and each of the two systems.

As predicted, the peak accuracy performance is observed in the clean/clean set. These values correspond to the best-case scenario and represent the baseline performance. The CHF approach outperforms the GMM approach with 8 % less error.

In the second set of experiments our working hypothesis (and literature experience) is confirmed in that all systems showed a significant decrease in the performance due to the mismatched conditions in classifying wild data while trained on clean audio. We interpret the fact that our tested GMM system performed with the lowest error rate as an indicator of quality for this particular system. The CHF system also decreased its performance significantly more than the GMM method.

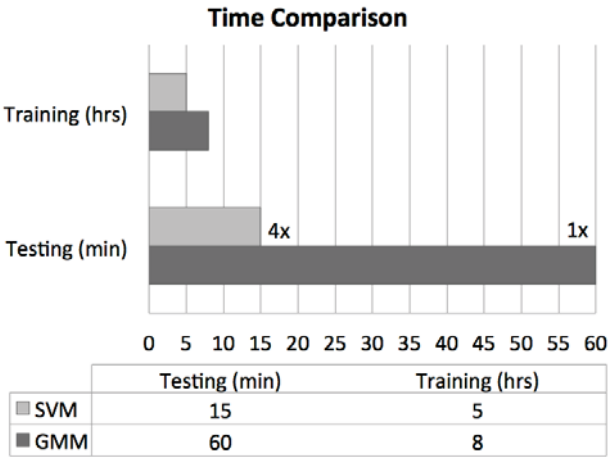

Figure 4. Bottom-Line time requirements for the experiments.

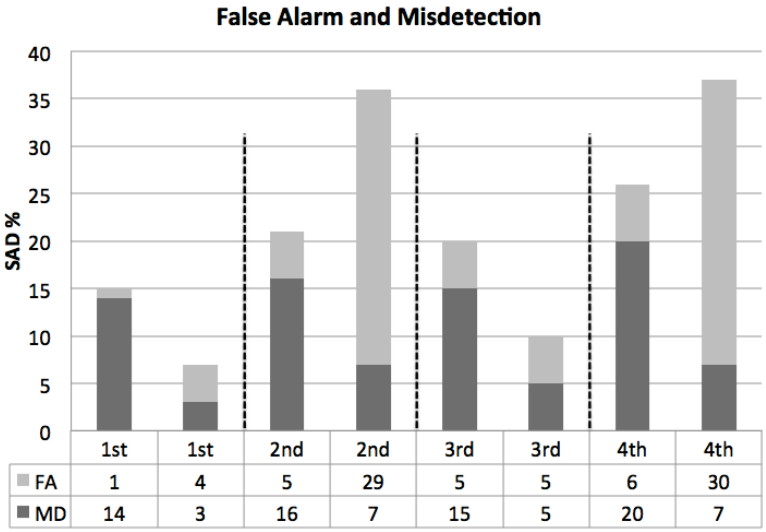

Figure 5. Two error types of the four sets of experiments for SHOUT and CFH.

The third set of experiments serves to compare that mismatched conditions do not have such an impact on accuracy when classifying clean data trained on wild data. We assume that wild data is more representative of the clean case than the clean data is of the wild audio. This has been often assumed in literature and is one of the principal assumptions in the "Big Data" movement: With enough data, machine learning algorithms perform better in general.[15] Our results indicate that this is indeed true. The overall performance of both systems is better here than in the second set. The CHF had 10 % less error than the GMM counterpart which we interpret as a limitation of the GMM approach for "wild" audio classification.

However, wild data type seems to be hardly representative of itself, confirming the principle assumption of this article for a need of cross-domain research in speech/non-speech detection. The last set of experiments, training on wild and testing on wild, lead to the lowest performance for the three systems. The GMM outperformed the CHF system in classifying wild data.

The Figure 5 shows the two error types of the four sets of experiments for each of the three systems. In general, SHOUT yielded a lower FA and the CHF a lower MD. The comparison between the first set, which is the baseline, and the fourth set, which is the wild/wild set, returned an increment of both errors, specially the FA.

Regarding computational demands, the SHOUT system was segmenting the audio in real time, which is at least 4 times slower than the CHF system. Each system had specific stages where data was processed at a slower rate. SHOUT algorithm was slow at iteratively retraining the GMMs after finding homogeneous models. The CHF system contained no slow points during the testing stage. However, the main time concern of this algorithm, the creation of the codebook and the SVM model training, were not considered which would have provided a slow point.

We used a state-of-the-art GMM-based system that was optimized to deal with noise and mismatched conditions. As a result its performance in the clean/clean case was suboptimal. However, its performance degraded the least for the other three experimental sets, making a case for being a quite stable approach. As discussed previously, the major disadvantage of GMM segmentation is that the models need to be trained on a matching dataset. If the acoustic characteristics of the audio under evaluation are too different from the characteristics of the training data, the accuracy of the segmentation will be poor.

The CHF system performed comparably well to SHOUT in clean data. Despite what it was designed for and based on our results and the literature (see Section 2), the CHF using SVM classification do not seem to cope well with the task of classifying wild data. The advantage of using a codebook system for this study is that it allows us to analyze the generated code entries and their resulting use. It turns out that all codes ended up

being very similar, but at the same time very few codebook entries were used with high frequency. This leads us to further confirm the conclusion that current SVM approaches may not to be promising for this task. Note the consistency of the CHF system when testing the clean sets regardless of the training nature. This was because the codebook was compensating the variability of the training sets.

7. CONCLUSIONS AND FUTURE WORK

Speech/non-speech detection and audio segmentation are important front-end tasks of higher level audio processing. With wild data becoming the main source of multimedia information from the internet, the importance of being able to accurately segment such cross-domain data is becoming more relevant. Currently, little research has been done to analyze the audio segmentation task for consumer-produced media. With a completely generic, cross-domain learning algorithm being unavailable, this paper presented a different approach to the task, which performed 50 % better and 4 times faster on meetings data but underperformed by 44 % on consumer-produced data in contrast to the GMM approach.

An analysis on the performance of the new approach was presented with the purpose of exploring a different method and more important, for a systematic understanding of the challenges of audio classification of consumer-produced videos. For future work, we would like to verify our hypothesis that a generic codebook is the right choice for speech only detection, or if it's better suitable for multiclass detection. Following this line of research we would investigate the type of sounds that each of the clusters from the codebook represent, and if a multiclass segmentation for example, speech/non-speech/music, would be more suitable for the generic codebook. It would also be insightful to make a detailed analysis of the FA and MD segmenting errors for each system to identify and understand more strengths and weaknesses. Furthermore, most of the work that reports results on wild data is inspired by the speech literature and uses tools originally developed in the speech community. As a result, MFCC features, with their restricted characteristics that suppress valuable information of the audio signal, are used. Investigating alternative features would therefore be a very valid line of work.

ACKNOWLEDGMENTS

Thanks to the original idea and help from PhD. Bhiksha Raj, the advice of Paola Garcia and PhD. Juan Arturo Flores Nolazco.

REFERENCES

[1] Google, "Frequently Asked Questions," (2011).

[2] Hain, T. and Woodland, P. C., "Segmentation and classification of broadcast news audio," in [*Proceedings of ICSLP*], 2727–2730 (1998).

[3] Istrate, D., Scheffer, N., and Fredouille, C., "Broadcast news speaker tracking for ester 2005 campaign," in [*In Proc. Interspeech 2005*], (2005).

[4] Saunders, J., "Real time discrimination of broadcast speech music," in [*In Proceedings of IEEE ICASSP 1996*], 993–996 (1996).

[5] Lu, L., Li, S., and Zhang, H., "Content-based audio segmentation using support vector machines," in [*In Proceedings of IEEE ICME 2001*], 749–752 (2001).

[6] Martin-Iglesias, D., Bernal-Chaves, J., Pelaez-Moreno, C., Gallardo-Antolin, A., and de Maria, F. D., "A Speech Recognizer based on Multiclass SVMs with HMM-Guided Segmentation," **3817** (2005).

[7] DARPA, "Robust Automatic Transcription of Speech," (2011).

[8] Ng, T., Zhang, B., Nguyen, L., Matsoukas, S., Zhou, X., Mesgarani, N., Vesely, K., and Matejka, P., "Developing a Speech Activity Detection System for the DARPA RATS Program," in [*Proceedings of Interspeech*], (2012).

[9] Misra, A., "Speech/Nonspeech Segmentation in Web Videos," in [*Proceedings of Interspeech*], (2012).

[10] Huijbregts, M., *Segmentation Diarization and Speech Transcription: Surprise Data Unraveled*, PhD thesis, Universiteit Twente (2008).

[11] Anguera Miro, X., Bozonnet, S., Evans, N., Fredouille, C., Friedland, G., and Vinyals, O., "Speaker diarization: A review of recent research," *IEEE Transactions on Audio, Speech, and Language Processing* **20**(2), 356–370 (2012).

[12] Elizalde, B., "Segment and Conquer," (2012). 2nd Multimedia and Vision Meeting in Greater New York Area.

[13] NIST, "Rich transcription 2006 spring meeting recognition evaluation plan v2," in [*In Rich Transcription 2006 Meeting Recognition Workshop*], (2006).

[14] et al, K. M., "Multivariate Analysis. Academic Press," (1979).

[15] Halevy, A., Norvig, P., and Pereira, F., "The unreasonable effectiveness of data," *Intelligent Systems, IEEE* **24**(2), 8–12 (2009).