A DIVIDE-AND-CONQUER APPROACH TO LATENT PERCEPTUAL INDEXING OF AUDIO FOR LARGE WEB 2.0 APPLICATIONS

Shiva Sundaram
Deutsche Telekom Laboratories,
Quality and Usability Lab,
TU-Berlin, Berlin, Germany.
shiva.sundaram@telekom.de

Shrikanth Narayanan
Signal Analysis and Interpretation Lab (SAIL),
Dept. of Electrical Engineering-Systems,
Univ. of Southern California (USC), Los Angeles, USA.
shri@sipi.usc.edu

ABSTRACT

In the recently proposed latent perceptual indexing of audio, a collection of clips is indexed using unit-document frequency measures between a set of *reference* clusters as units and the clips as the documents. The reference units are derived by clustering the bag-of-feature vectors extracted from the whole audio library using an unsupervised clustering technique. Indexing is achieved through reduced-rank approximation (using singular-value decomposition) of the unit-document co-occurrence measure matrix that is obtained for the given set of reference clusters and the collection of audio clips. In our initial investigation, the k-means algorithm was used to derive the reference units. In this paper, we attempt to reduce the computation load requirements for the k-means algorithm and singular-value decomposition by randomly splitting the training data into smaller sized parts instead of working on it as a whole. We present results of classification experiments on the BBC sound effects library and our results indicate this approach can significantly reduce the computation time without significant loss in classification performance.

***Index Terms*—** content based audio retrieval, audio classification, clustering, Latent Perceptual Indexing, Web 2.0 applications.

1. INTRODUCTION

Present day Web 2.0 applications allow users to create, upload, edit and access multimedia documents. To provide efficient, automatic access to the users, the back-end indexing system needs to categorically organize the documents. To have full flexibility, indexing using tags/captions (for text-based queries) and the content (for example-based queries) is desirable. Although document indexing is an off-line process, due to large number of users and even larger number of documents, the computational load [1] required for indexing can be very high. This aspect becomes even more critical for indexing using the content instead of the tags/captions because the information extraction bandwidth for multimedia documents can be significantly larger. In light of this issue, the work presented in this paper focuses on reducing the computational load involved in Latent Perceptual Indexing of audio clips for example-based query and retrieval.

Latent Perceptual Indexing (LPI) [1, 2], seeks a single vector representation of an audio clip within a collection by using unit-document frequency measures. This approach is analogous to Latent Semantic Indexing (LSI)[3, 4]. The units or reference clusters in LPI are taken to be equivalent to terms (or words) in LSI and documents are the audio clips that are equivalent to text documents in LSI. As shown in [1], this results in a single, low-dimensional vector and the similarity between the clips can be derived through vector distance measure. The main advantage of this approach is that it allows for comparison of arbitrary audio clips through vector similarity measure that also embodies both semantic and perceptual similarities [2].

In this data-driven approach, there are two computationally intensive steps involved in obtaining a latent representation for the audio clips: the $k-$means algorithm used to discover the *units* or reference clusters, and singular value decomposition (SVD) of the unit-document frequency matrix to obtain the reduced-rank approximation. The main focus of this paper is to reduce the computational load for latent perceptual indexing. Here, we investigate a *divide-and-conquer* approach of splitting the available training data into smaller, manageable parts and then processing them individually. It is easy to see that such an approach reduces the computational load required for indexing as (1) clustering and SVD algorithms run faster on smaller data sets, (2) it allows for parallel computations and (3) the runtime memory requirements are significantly lower.

The organization of this paper is as follows: related work within the context of content-based retrieval is presented next followed by the details of the experiments performed and the results. Finally, the paper concludes with a discussion of the results and future directions.

1.1. Related Work

Content-based retrieval of audio mainly involves organization and classification according to the semantic labels. Typical examples [5, 6, 7] are those that use category based modeling for a selection of audio clips. In [5, 6] the system is evaluated on a library of categories such as *animals, bells, crowds, female, laughter, machines, male voices, percussion instruments, telephone, water sounds* etc. In contrast to this, LPI proposed in [1] attempts to characterize whole, unstructured [2] audio clips. Related work in this domain of modeling and retrieval for unstructured audio are [8, 9, 10]. In [8], the author improves the labeling scheme by creating a mapping from each node of a hierarchical model in the abstract semantic space to the acoustic feature space. The nodes in the hierarchical model (represented probabilistically as words) are mapped onto their corresponding acoustic models. In [9], the authors propose a similar approach of modeling features with semantic text labels in the captions. In [10], their approach uses semantic relations in language. Here the authors have used WordNet to generate words for a given audio clip using acoustic feature similarities, and then retrieve clips that are similar to the tags. While these viable methods have been successful, as far as we know, strategies to extrapolate them to large data sets have not been explored or presented. In this respect, the main

[1] computational load refers to memory and cpu time requirements.

[2] the term *unstructured* refers to audio clips recorded in any scene that may contain any number of unknown sources that may also overlap temporally without any well defined segmentation points. Such clips do not belong to any pre-specified domain or category.

978-1-4244-4291-1/09/$25.00 ©2009 IEEE

Authorized licensed use limited to: University of Southern California. Downloaded on September 21, 2009 at 22:23 from IEEE Xplore. Restrictions apply.

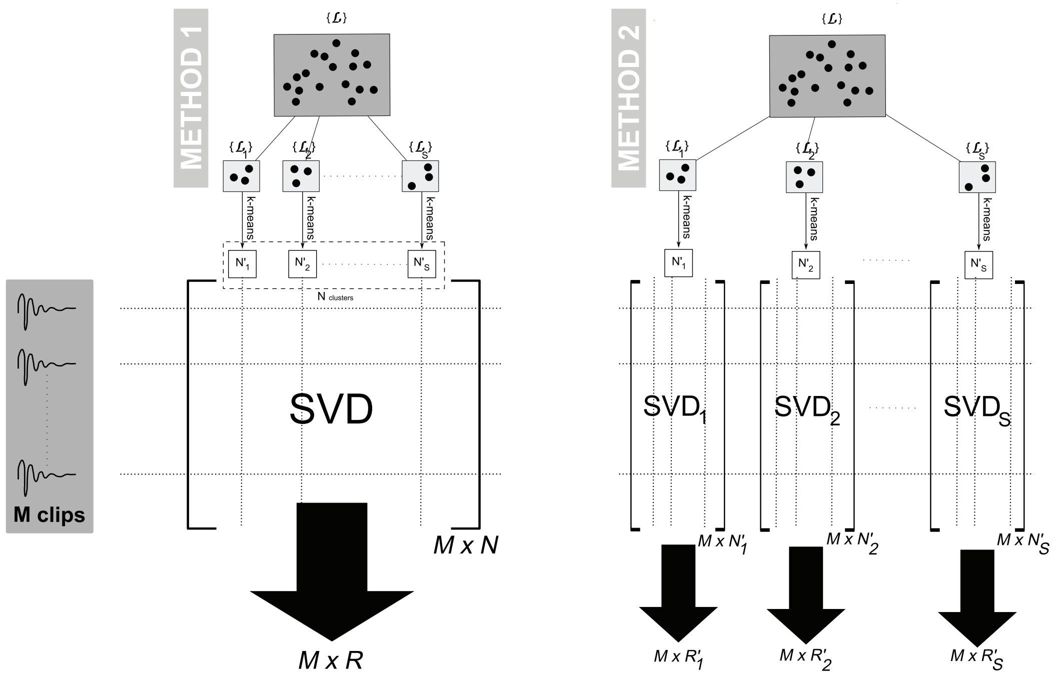

Fig. 1. Two approaches to Latent Perceptual Indexing by splitting the data $\{\mathcal{L}\}$. **METHOD 1:** clustering individual parts and then performing combined SVD. **METHOD 2:** clustering individual parts and performing individual SVDs, and then augmenting the respective vectors.

contribution of the work presented here is to investigate methods to reduce the computational load for LPI by a divide-and-conquer approach that is suitable for large-scale parallel computations.

2. PROPOSED APPROACH

Given a collection of M audio clips, the procedure to derive a latent perceptual representation is as follows. First, a *bag of feature-vectors* $\mathcal{L}$ is extracted from the M clips. Using an unsupervised clustering technique, the set of features $\mathcal{L}$ is clustered into N distinct reference clusters (whose centroids are the reference points). Then each feature vector extracted from say the m^{th} clip is quantized into one of the N clusters, and a N dimensional unit-document frequency measure is estimated [1, 2]. Repeating this procedure for the collection of M clips results in a $F_{M \times N}$ matrix representation of the library. By SVD, a reduced rank ($\tilde{F}_{M \times R}$ where $R < N$) approximation of this sparse representation is obtained resulting in mapping audio clips to points in a latent perceptual space. Thus each audio clip is represented as a single vector in a latent space spanned by the eigen vectors of the SVD step. Using a similar procedure, a clip outside the collection of M clips can also be represented in this latent space and similarity between two clips is the vector dot product of their corresponding vectors. The motivation for Latent Perceptual Indexing method and the vector similarity measures are discussed in greater detail in [1].

In our initial approach, we presented the retrieval (and classification) performance of the system as a function of number of reference clusters thus derived. In this case, the $k-$means algorithm was applied to the whole set $\mathcal{L}$ of P feature vectors ($P \approx 2.5 \times 10^6$ feature vectors). In this work, to easily manage such large data sets and facilitate parallel computations we propose to split the available data $\mathcal{L}$ into S parts, and then combine them to index the audio clips. By this divide-and-conquer approach we can obtain $N' = \frac{N}{S}$ smaller number of clusters for the individual parts and then use the resulting $N' \cdot S$ clusters to derive the unit-document frequency matrix. In this work we present performance analysis of the retrieval system using this approach and compare it to our earlier approach of clustering the whole set $\mathcal{L}$ into N clusters. Two possibilities are considered:

1. The available training data $\mathcal{L}$ is split into S parts, and the smaller individual parts $\mathcal{L}_1, \mathcal{L}_2, \ldots \mathcal{L}_S$ are clustered in to $N'_1, N'_2, \ldots, N'_S$ where $N'_i = \frac{N}{S} \forall i \in \{1, 2, \ldots, S\}$. Then, the sets of reference clusters N'_i are combined (resulting in N clusters again) for the subsequent unit-document frequency matrix and mapping into the latent perceptual space. In this case, the $k-$means algorithm is performed on the smaller $\mathcal{L}_1, \mathcal{L}_2, \ldots \mathcal{L}_S$ parts but the SVD is performed only once. Therefore, while it has reduced the computational load of the $k-$ means algorithm, the SVD still needs to be performed on an $M \times N$ matrix. Note that a R dimensional representation is obtained for the audio clips.
2. Using the same sets of $N'_1, N'_2, \ldots, N'_S$ clusters it is also possible to derive S individual term-document frequency matrices, then perform S SVDs and then combine the resulting vectors by augmenting them. In this case, S $k-$ means algorithms are performed on $\mathcal{L}_1, \mathcal{L}_2, \ldots \mathcal{L}_S$ and then S SVDs are performed respectively. The S individual SVDs however, are performed on their respective $M \times N'_i$ matrices where $N'_i = \frac{N}{S} \forall i \in \{1, 2, \ldots, S\}$. Due to augmentation, this results in a $R = R'_1 + R'_2 + R'_3 \ldots R'_S$ dimensional vectors for the audio clips.

These two possible methods are illustrated in figure 1. Splitting the data into smaller sets and then combining them does not change the original motivation or formulation of Latent Perceptual Indexing. However, the information extracted and the eventual representation in the latent space is different in this case. The difference

Authorized licensed use limited to: University of Southern California. Downloaded on September 21, 2009 at 22:23 from IEEE Xplore. Restrictions apply.

Category	Files	Category	Files
IMPACT	16	NATURE	85
OPEN	8	SPORTS	151
TRANSPORTATION	295	HUMAN	357
AMBIENCES	311	EXPLOSIONS	18
MILITARY	102	MACHINERY	117
ANIMALS	359	SCI-FI	121
OFFICE	144	POLICE	96
HORROR	98	PUBLIC	44
AUTOMOBILES	53	DOORS	4
MUSIC	25	HOUSEHOLD	38
ELECTRONICS	49		

Table 1. Distribution of clips under each semantic category.

stems from the fact that perceptually descriptive units derived by $k-$means on the smaller parts does not equal the units that would have been obtained by directly clustering $\mathcal{L}$. The set-wise intersection of N reference clusters or points derived through $k-$means clustering of $\mathcal{L}$ with the N reference clusters derived by individually clustering $\mathcal{L}_1, \mathcal{L}_2, \ldots \mathcal{L}_S$ may or may not be null. These however are differences caused by the practical aspects of clustering that also projects to the unit-document co-occurrence measure subsequently calculated. As mentioned earlier, the advantage of splitting the data set into smaller parts is that the computations for the individual sets can be done in parallel and the resources required by the computationally intensive $k-$means and SVD on smaller data sets are many orders less in comparison to directly working on *all* the data together. Consider the savings in method 1: the overall worst-case runtime for $k-$mean algorithm for N clusters of P points of d dimensional data is $\mathcal{O}\left(NPd\right)$ [11]. If we split the P points into (say) S equal parts, then we only need to find $\frac{N}{S}$ clusters from each of the $\frac{P}{S}$ points. If the S parts are clustered in parallel, this reduces the runtime to $\mathcal{O}\left(\frac{NPd}{S^2}\right)$. Similarly, consider runtime savings in method 2: SVD runtime for sparse $M \times N$ matrix with c non-zero elements per column is $\mathcal{O}\left(MNc\right)$, by splitting the data and running in parallel the runtime is $\mathcal{O}\left(M\frac{N}{S}c\right)$. This form of runtime reduction is relevant for very large data sets that is prevalent in Web 2.0 applications.

2.1. Data

For the framework presented, 2,491 whole audio clips from the BBC Sound Effects Library [12] were used. The selection of the clips used here belong to twenty one high-level semantic categories and the number of clips under each category is shown in table 1. For feature extraction, all the 44.1kHz stereo clips were down-sampled to 16.0kHz and converted to mono.

2.2. Acoustic Features

For the experiments, a representation for an audio clip in the latent perceptual space is derived using the signal-level features. In this work, a fourteen dimensional feature-vector set popular in content-based audio classification tasks is extracted from each audio clip. Twelve dimensions are comprised of perceptually motivated Mel-frequency cepstral coefficients (MFCCs), with the remaining two representing the spectral centroid (SC) and spectral-roll-off frequency (SRF) measures. The SC and SRF measures are normalized to half the sampling frequency of the audio clips. The features are extracted by windowing the audio signal with a twenty millisecond Hamming window at every ten millisecond.

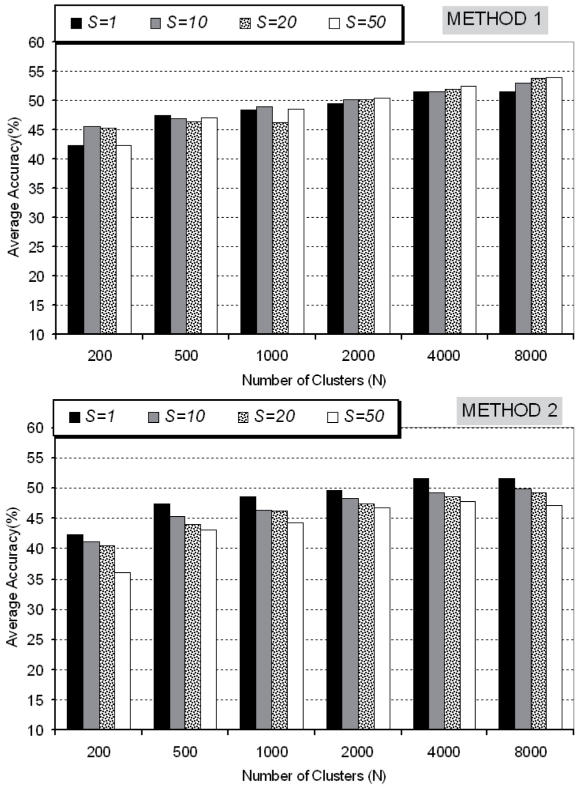

Fig. 2. Average Classification Performance using 3 nearest neighbor rule on the twenty one sound categories.

3. EXPERIMENTS AND RESULTS

All the classification/retrieval performances have been estimated by averaging results from 10-fold cross-validation. First, the available M clips are randomly split into 10 equal parts. For each fold, one of the 10 parts was used as the test set and the remaining nine were used for training. Note that the $k-$means and unit-document frequency measures were performed only on the train set.

Similar to our initial investigation in [2], we target the performance estimates for $N = 200, 500, 1000, 2000, 4000, 8000$ reference clusters. For the data splitting strategies discussed here, the training data (in each fold of the cross-validation) was split into $S = 1$ (no splitting) $S = 10, 20, 50$ parts. The number of clusters N' for each of the S parts are $N' = \frac{N}{S}$. The average classification accuracy for the twenty one semantic labels are shown in figure 2. The classification was performed using the 3 nearest neighbor $(3NN)$ rule. The average chance-level classification performance is 11.4%

For *Method 1*, for the original case of $S = 1$ the average accuracy improves with larger number of clusters, with best performance in $N = 4000$. For $S = 10, 20, 50$, the average accuracy is comparable to $S = 1$ and for $N = 4000, 8000$ it is higher by about 2% compared to the $S = 1$ case. The reason for observing slightly better performance could be related to the practical aspects of the $k-$means algorithm: For LPI, smaller number of reference centroids derived from the smaller sets of data could be better than discovering large number of clusters from large number of feature points. For the case presented here, $k-$means can also be seen as an unsupervised model fitting procedure that results in a *better fit* for the smaller size of feature points. Best performance is for $S = 50$ and $N = 8000$. The

Authorized licensed use limited to: University of Southern California. Downloaded on September 21, 2009 at 22:23 from IEEE Xplore. Restrictions apply.

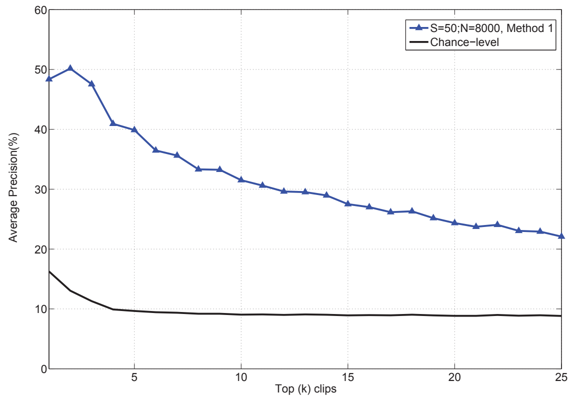

Fig. 3. Average precision for the top 25% of the retrieved clips.

average precision and recall retrieval rates for this case is given in figure 3. The chance-level retrieval performance of randomly choosing clips is also included.

For *Method 2*, the increasing trend in average accuracy values as for higher values of N is again observed. Additionally, the average classification performance for $S = 10, 20, 50$ cases are consistently poorer compared to the $S = 1$ case of no data splitting. An interesting result is that amongst the three splitting cases, the classification accuracy for $S = 10 > S = 20 > S = 50$. This indicates that performing smaller SVDs and then augmenting the vectors results in a poorer representation of the clips in the augmented latent space. Given the performance improvement due to splitting in Method 1, the poorer performance here can be attributed to the SVD dimension reduction. Since, in this case, the unit-document information measure from only a part of the set of reference units is used at a time. Additionally, in this case the global weighting factor for the unit-document matrix [4] considers only $\frac{N}{S}$ factors at a time that could cause un-normalized scaling of the columns of the unit-document matrix. Although the performance is degraded this approach of performing multiple, smaller SVDs can be more practical for very large collections that can be expected especially in Web 2.0 applications where $M \approx 10^5$ to 10^6 and $N \approx 10^4$ to 10^5.

4. CONCLUSION AND FUTURE WORK

In this work, an alternative, *divide-and-conquer* approach to latent perceptual indexing has been presented. The two computationally intensive processing steps: the $k-$means algorithm for discovering reference units and the SVD for rank-reduction of the unit-document frequency matrix, have been approached in a computationally more manageable manner by diving the available data set into smaller parts and processing them individually. This is compared to our original approach of processing the complete data together within the context of content-based audio classification experiments on the BBC sound effects library. Our results indicate that this is indeed a viable approach to handling large amounts of data that is typical in Web 2.0 applications. The performance of the latent perceptual indexing framework is comparable to other retrieval systems that targets unstructured audio indexing [13, 14].

The promising results of this work allows us to look at other strategies for discovering reference units that is critical in latent perceptual indexing of audio. Similar to probabilistic Latent Semantic Analysis (PLSA) [15] as a part of our future work, we plan to explore the potential of probabilistic modeling, and other strategies to rank or weight the reference units that will result in a more efficient representation of the acoustic information.

5. REFERENCES

[1] S. Sundaram and S. Narayanan, "Audio Retrieval by Latent Perceptual Indexing," *IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP), Las Vegas, USA.*, 2008.

[2] S. Sundaram and S. Narayanan, "Classification of sound clips by two schemes: Using Onomatopoeia and Semantic labels," *2008, International Conference on Multimetia and Expo (ICME), Hannover, Germany.*, pp. 1341–1344, June 2008.

[3] S. Deerwester, S. T. Dumais, G. W. Furnas, T.K. Landaurer, and R. Harshman, "Indexing by Latent Semantic Analysis," *Journal of the American Society for Information Science*, vol. 6, no. 41, pp. 391–407, 1990.

[4] J.R.Bellegarda, "Latent Semantic Mapping," *IEEE Signal Processing Magazine*, vol. 22, pp. 70–80, September 2005.

[5] E. Wold, T. Blum, D. Keislar, and J.W Wheaton, "Content-Based Classification, Search, and Retrieval of Audio," *IEEE Multimedia*, vol. 3, no. 3, pp. 27–36, Fall 1996.

[6] G.Guo and S. Z. Li, "Content-Based Audio Classification and Retrieval by Support Vector Machines," *IEEE Transactions on Neural Networks*, vol. 14, no. 1, pp. 209–215, January 2003.

[7] L. Liu, H. J. Zhang, and H. Jiang, "Content Analysis for Audio Classification and Segmentation," *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 7, pp. 504–516, October 2002.

[8] M. Slaney, "Semantic Audio Retrieval," *International Conference on Acoustic Speech and Signal Processing (ICASSP), Orlando, USA.*, pp. 13–17, May 2002.

[9] L. Barrington, A. Chan, and D. Turnbul land G. Lanckriet, " Audio Information Retrieval using Semantic Similarity," *IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP), Honolulu, Hawaii, USA.*, vol. 2, pp. II–725–II–728, 2007.

[10] P. Cano, M. Koppenberger, S. LeGroux, J. Ricard, P. Herrera, and N. Wack, "Nearest-Neighbor Generic Sound Classification with a WordNet-based Taxonomy," *In Proceedings* 116^{th} *Audio Engineering Society (AES) Convention, Berlin, Germany.*, 2004.

[11] T. Kanungo, D. M. Mount, N. Netanyahu, C. Piatko, R. Silverman, and A. Y. Wu, "An efficient k-means Clustering Algorithm: Analysis and implementation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, pp. 881–892, 2002.

[12] "The BBC Sound Effects Library Original Series," *http://www.sound-ideas.com*, May 2006.

[13] D. Turnbull, L. Barrington, D. Torres, and G. Lackreit, "Semantic Annotation and Retrieval of Music and Sound Effects," *IEEE Transactions on Audio, Speecn and Language Processing*, vol. 16, pp. 467 – 476, February 2008.

[14] G. Chechik, E. Ie, M. Rehn, S. Bengio, and D. Lyon, "Large-scale content-based audio retrieval from text queries," *Proceeding of the 1st ACM international conference on Multimedia information retrieval*, pp. 105–112, 2008.

[15] T. Hoffman, "Probabilistic latent semantic indexing," *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in Information Retrieval*, pp. 50–57, 1999.

Authorized licensed use limited to: University of Southern California. Downloaded on September 21, 2009 at 22:23 from IEEE Xplore. Restrictions apply.