The IIT-B Query-by-Example System for MediaEval 2015

Hitesh Tulsiani, Preeti Rao
Department of Electrical Engineering,
Indian Institute of Technology Bombay, India
{hitesh26, prao}@ee.iitb.ac.in

ABSTRACT

This paper describes the system developed at I.I.T. Bombay for Query-by-Example Search on Speech Task (QUESST) within the MediaEval 2015 evaluation framework. Our system preprocesses the data to remove noise and performs subsequence DTW on posterior/bottleneck features obtained using four phone recognition systems to detect the queries. Scores from each of these subsystems are fused to get the single score per query-utterance pair which is then calibrated with respect to the cross entropy evaluation metric.

1. INTRODUCTION

The goal of the QUESST task within the MediaEval 2015 framework is to determine the presence of a spoken query in an unlabeled speech data set by building a language independent system. In this year's QUESST task, the data consisted of about 18 hours of noisy audio from 7 different languages. More details about the task can be found in [1].

To minimize the effect of noise, we preprocess our data (both the queries and utterances) and follow it with speech activity detection to remove silence frames. Our approach, to solve the task, is inspired by Hazen et al.[2]. A block-diagram of our system is shown in Figure 1 and is inspired by [3].

2. SYSTEM DESCRIPTION

2.1 Preprocessing - Noise Removal

We use spectral subtraction to remove noise from the audio. Power spectral density (PSD) of noise is estimated using the minimum statistics technique described by R. Martin [4]. The technique used to estimate noise PSD makes the assumption that during speech pause or within brief periods in between words the speech energy is close to zero. Thus, by tracking the minimum power within a finite window large enough to bridge high power speech segments the noise floor can be estimated. We next remove the silence at the start and end of an utterance using a simple energy based speech activity detector.

MediaEval 2015 Workshop, Sept. 14-15, 2015, Wurzen, Germany

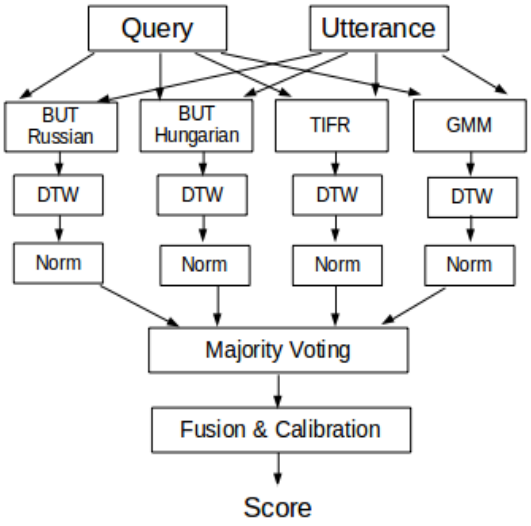

Figure 1: Block diagram of the IIT-B system

2.2 Subsystems

We make use of 4 subsystems:

1. Two DNN based phone recognisers (Hungarian and Russian) trained on the SpeechDat-E corpus by Brno University of Technology (BUT)[5]. These are used to extract posterior and bottleneck features.

2. A phone recogniser trained on Hindi database [6] (referred to as TIFR phone recogniser from here on). TIFR phone recogniser is MLP based and is trained using 39 dimensional MFCC features. It has a single hidden layer with 700 neurons and 36 output neurons. We extract phone posteriors using the TIFR phone recogniser.

3. 64-GMM system trained in unsupervised manner on QUESST-2015 database using 36 dimensional MFCC features [7] (the energy of the audio was not used as feature because large energy variations were observed across utterances). We used this system to extract Gaussian posteriorgrams.

2.3 DTW

We use the standard subsequence DTW as implemented in [3]. The query is allowed to start at any frame of the test

	eval		dev	
Query Type	actCnxe/minCnxe	ATWV/MTWV	actCnxe/minCnxe	ATWV/MTWV
T1	0.9330/0.9117	0.0531/0.0661	0.8971/0.8680	0.1434/0.1449
T2	0.9852/0.9637	-0.0099/0.0178	0.9214/0.9113	0.0492/0.0528
T3	0.9313/0.9109	0.0525/0.0627	0.9348/0.9210	0.0454/0.0461
overall	0.9536/0.9364	0.0254/0.0421	0.9213/0.9082	0.0812/0.0816

Table 1: Overall and per query type (T1/T2/T3) summarization of results on evaluation and development datasets.

utterance and the locally optimal detection is the one that has the smallest accumulated distance. Also, to avoid the preference for the shorter paths, accumulated distances are normalized by the corresponding detected path lengths. For distance measure, we have used Pearson product-moment correlation for bottleneck features (BUT - Hungarian and Russian) and inner product for posteriors (BUT - Hungarian and Russian, TIFR, 64-GMM). A filtering step is then applied to remove detected candidates which are very large or very small in duration compared to the query length.

2.4 Fusion and Calibration

Our approach is most similar to the discriminative fusion approach proposed by A. Abad et al. [8]. Scores are first normalized to zero mean and unit variance per query to allow for use of a single threshold. Then the detections are aligned and only those detections for which at least half the systems show overlap in time are retained (majority voting) to reduce the false alarms. This leaves us with multiple detections of a query in an utterance. So for each query-utterance pair we will get multiple score vectors (A score vector is a collection of scores from all the subsystems for a possible detection of query in an utterance). Our score vector has six elements (BUT Hungarian-Posterior and Bottleneck, BUT Russian-Posterior and Bottleneck, TIFR - Posterior, GMM - Posterior).

Since the task requires to give only one score per query-utterance pair, we determine best score vector per query-utterance pair using a two-step procedure:

1. First step is inspired by Hazen et al.[2]. Scores from various subsystems $S(X|K_i)$ are combined according to equation:

$$S(X|K_1K_2...K_N) = -\frac{1}{\alpha}log(\frac{1}{N}\sum_{i}^{N} exp(-\alpha S(X|K_i))) \quad (1)$$

   where varying α between 0 to 1 changes the averaging function from geometric mean to arithmetic mean (we have used $\alpha = 1$).

2. In the second step, we make use of the combined score obtained in first step to determine the best candidate for an utterance. We retain the individual scores of the subsystems along with the combined score (obtained using equation 1) corresponding to the best detected candidate, thus giving us one score vector per query-utterance pair.

All of these score vectors (corresponding to different query-utterance pairs) are then used to train a binary logistic classifier [3] which gives us the fused score representative of query-utterance pair. The fused scores are then calibrated with respect to cross entropy evaluation metric to give us log-likelihood score.

3. RESULTS AND DISCUSSION

Table 1 shows our results for development and evaluation queries. Probably due to the high amount of noise (and reverberation) in the dataset, the overall cross-entropy score is poor even after noise removal. If we look at the scores for each query type, clearly our system works best for the T1 query type. This can be attributed to the fact that we didn't take any special steps to counter T2 and T3 query types like word level reordering (for T2 queries) and partial matching (for T3 queries). Also, we didn't calibrate our score for Term Weighted Values (TWV) resulting in very low ATWV/MTWV scores.

We observed that after subsequence DTW many possible detections (candidates) were found for a query in an utterance. This clearly suggests that posteriors and bottleneck features used were not robust enough for the given noisy and multilingual data. Also, we rely heavily on our first step of fusion which is nothing but the arithmetic mean of scores (since $\alpha = 1$) from various subsystems to detect the best candidate for a given query-utterance pair. So a high score from even one of the subsystems can make the combined score (obtained after Step 1 of fusion) biased towards it, leading to the selection of that candidate over other candidates with moderate scores from all the systems.

Our experiments were done on a computer with Intel i7-4790 CPU (3.60GHz, 8 cores), 16GB RAM. For searching, all the posteriorgrams for a query-utterance pair were loaded in memory. This caused high memory usage for longer utterances (Peak memory usage of around 15GB). It took us around 80 hours to search approximately 475 seconds of query in 18 hours of audio database per subsystem, leading to SSF of 0.0093 per sec.

4. CONCLUSION

We have described the system developed at IIT-B for QUESST task. To combat the effect of noise in data, we used spectral subtraction. Spectral subtraction reduces noise but is also known to create artifacts in speech and so posteriors/bottleneck features were not robust enough for the given noisy and multilingual data. It would be interesting to study the performance of our system without noise suppression. The main novelty of our work was a two-step fusion approach where in the first step we decide the best candidate for a query-utterance pair and in the second step we train a logistic regression classifier. The effect of the first step of fusion for different values of α on the cross entropy score needs to be investigated.

5. REFERENCES

[1] Igor Szőke, Luis J. Rodriguez-Fuentes, Andi Buzo, Xavier Anguera, Florian Metze, Jorge Proenca, Martin Lojka, and Xiao Xiong. Query by example search on speech at mediaeval 2015. In *Working Notes Proceedings of the Mediaeval 2015 Workshop*, 14-15 September 2015.

[2] T. Hazen, W. Shen, and C. White. Query-by-example spoken term detection using phonetic posteriorgram templates. In *Proc. ASRU*, 2009.

[3] Igor Szőke, Lukáš Burget, František Grézl, Jan Černocký, and Lucas Ondel. Calibration and fusion of query-by-example systems - BUT SWS 2013. In *Proc. ICASSP*, pages 7899–7903, 2014.

[4] R. Martin. Noise power spectral density estimation based on optimal smoothing and minimum statistics. *IEEE Transactions on Speech and Audio Processing*, 9(5):504–512, 2001.

[5] F. Grezl, M. Karafiat, S. Kontar, and J. Cernocky. Probabilistic and bottle-neck features for LVCSR of meetings. In *Proc. ICASSP*, pages 757–760, 2007.

[6] V. Chourasia, K. Samudravijaya, and M. Chandwani. Phonetically rich hindi sentence corpus for creation of speech database. In *Proc. O-Cocosda*, pages 132–137, 2005.

[7] Y. Zhang and J. Glass. Unsupervised spoken keyword spotting via segmental DTW on gaussian posteriorgrams. In *Proc. ASRU*, pages 398–403, 2009.

[8] A. Abad, L. Rodríguez-Fuentes, M. Penagarikano, A. Varona, and G. Bordel. On the calibration and fusion of heterogeneous spoken term detection systems. In *Proc. INTERSPEECH*, pages 20–24, 2013.