

Speed @ MediaEval 2015: Multilingual Phone Recognition Approach to Query By Example STD

Alexandru Caranica, Andi Buzo, Horia Cucu, Corneliu Burileanu
Speed Research Laboratory, University Politehnica of Bucharest
Bucharest, Romania

alexandru.caranica@speed.pub.ro {andi.buzo, horia.cucu, corneliu.burileanu}@upb.ro

ABSTRACT

In this paper, we attempt to solve the Spoken Term Detection (STD) problem for under-resourced languages by a phone recognition approach within the Automatic Speech Recognition (ASR) paradigm, with multilingual acoustic models from six languages (Albanian, Czech, English, Hungarian, Romanian and Russian). The Power Normalized Cepstral Coefficients (PNCC) features are used for improved robustness to noise, along with Phone Posteriorgrams in order to obtain content-aware acoustic features as independent as possible from speaker and acoustic environment.

1. INTRODUCTION & APPROACH

We approach the Query by Example Search on Speech Task (QUESST [1]) @ MediaEval 2015 by using multilingual acoustic models (AM) trained with six languages (Albanian, Czech, English, Hungarian, Romanian and Russian). The task involves searching for audio content within audio content using an audio query.

The approach consists of two stages:

1. The indexing, i.e. the phone recognition of the content data
2. The searching, i.e. finding a similar string of phones in the indexed content that matches the one of the query by using a DTW based searching algorithm.

Unlike previous years, in 2015 the audio database features a more challenging acoustic environment, by introducing noise and reverberation. We expected PNCC features to perform better in this scenario.

As increasing the training database in comparison with last year would go beyond the context of the competition (which aims at low-resourced languages), we tried introducing new languages in the training phase to see how they perform, along with a neural network based phoneme recognizer, from BUT [2].

1.1 Acoustic models

In our approach, one thing we want to compare is the effect of using PNCC features vs MFCC, along with the improvements a robust phone recognizer based on neural network classifiers can bring to our STD task. For the first comparison, we have built four acoustic models, using internal audio resources for training, as described in Table 1. The AM training and the phoneme recognition are made in a conventional way, using Hidden Markov Models (HMMs), in CMU Sphinx [3].

We have built an AM for each language, (AM1 - AM3). AM1 is trained with 8.7 hours of read speech. We could have trained the Romanian AM with more data, but as we stated in the introduction, we wanted to have balanced training data among different languages, for an under-resourced task. AM2 is trained with 4.1 hours of Albanian read speech and broadcast news. AM3

is trained with 3.9 hours of native English read speech from the standard TIMIT database [4]. AM4 is trained with all the data from the three languages above, phonemes that are common in different languages were trained together, thus reducing the number of phonemes to 98. This was necessary to try and keep uncertainty as low as possible during the recognition phase. The identification of the common phonemes was made based on International Phonetic Alphabet (IPA) classification [5]. Two speech features types are used in this first approach: the common Mel Frequency Cepstral Coefficients (MFCC) and the Power Normalized Cepstral Coefficients (PNCC) [6].

Table 1. Training data for HMM approach

ID	Language	No. phones	Training data [h]
AM1	Romanian	34	8.7
AM2	Albanian	36	4.1
AM3	English	75	3.9
AM4	Multilingual common phones	98	16.7

For our second approach, we used phone posteriorgrams that are output by the robust phoneme recognizer from BUT. This phone recognizer uses a split temporal context (STC) based feature extraction, with neural network classifiers [7] to output phone posteriorgrams, while Viterbi algorithm is used for phoneme string decoding. We can use the output of this tool in our DTW search algorithm as input features, to do the matching.

In order to use additional languages to build features for the phoneme recognizer, we used the pre-trained systems available at [8] and described in Table 2.

Table 2. Trained systems used for STC approach

ID	Language	No. phonemes	WER[%]
AM5	Czech	45	24.24
AM6	Hungarian	61	33.32
AM7	Russian	52	39.27

The languages used for training the systems described in Table 2 are from the SpeechDat-E Eastern European Speech Database [9]. Another incentive to use these systems is the existence of trained non-speech events mapped to the following tokens, which should prove useful with these years challenging acoustic environment:

- “int” for intermittent noise
- “spk” for speaker noise
- “pau” for silent pause

The STC approach is based on the theoretical study that significant information about phoneme is spread over few hundreds milliseconds and that an STC system can process two

parts of the phoneme independently. The trajectory representing a phoneme feature can then be decorrelated by splitting them into two parts, to limit the size of the model, in particular the number of weights in the neural-net (NN). The system uses two blocks of features, for left and right contexts (the blocks have one frame overlap). Before splitting, the speech signal is filtered by applying the Hamming window on the whole block, so that the original central frame is emphasized. Dimensions of vectors are then reduced by DCT and results are sent to two neural networks. The posteriors from both contexts are, in the final stage, merged, after the front-end neural networks are able to generate a three-state per phoneme posterior model [10]. The above described features were used in this work as input to our search algorithm, which is described in the following section.

1.2 Searching algorithm

In ideal conditions, if an ASR system should output utterances with a 100% accuracy and precision, then the STD would be reduced to a simple character string search of a query within a textual content. As the experimental results show, we are far from the ideal case, hence we have to find within a content a string which is *similar* to the query.

Our proposed *DTW String Search* (DTWSS) uses the Dynamic Time Warping to align a string (a query) within a content. The search is not performed on the entire content, but only on a part of it by means of a sliding window proportional to the length of the query. The term is considered detected if the DTW scores above a threshold. This method is refined by introducing a *penalization* for the short queries and the spread of the DTW match. The formula for the score s is given by equation (1):

$$s = (1 - PhER)(1 + \alpha \frac{L_q - L_{Qm}}{L_{QM} - L_{Qm}})(1 + \beta \frac{L_w - L_s}{L_Q}) \quad (1)$$

where L_q is the length of the query, $L_{QM} = 18$ and $L_{Qm} = 4$ are the maximum and the minimum query lengths found in the development data set, L_w is the length of the sliding window, L_s is the length of the matched term in the content, while α and β are the tuning parameters. For this task, α and β are set to 0.6, from previous evaluations [12]. The penalizations in formula (1) are motivated by the assumption that for two queries of different length that match their respective contents by the same phone error rate (*PhER*), the match of the longer query is more probable to be the right one. Similarly the more compact DTW matches are assumed to be more probable than the longer ones. This algorithm is suitable for queries of type 1 and 2, because the DTW handles inherently the small variations from the query, but it is not suitable for queries of type 3 where word order may be inverted.

2. EXPERIMENTAL RESULTS

2.1 STD results

For our first comparison, the results obtained on the development database with the first two type of features (MFCC and PNCC) are shown in Table 3.

Table 3. MFCC vs PNCC performance comparison

ID	MFCC		PNCC	
	ACnxe	MinCnxe	ACnxe	MinCnxe
AM1	1.0061	0.9944	1.0061	0.9943
AM2	1.0059	0.9947	1.0058	0.9947
AM3	1.0055	0.9944	1.0047	0.9933
AM4	1.0047	0.9933	1.0047	0.9933

For the third type of features, posteriorgrams with STC approach, results are show in Table 4.

Table 4. Posteriorgram performance results

ID	ACnxe	MinCnxe
AM5	1.0055	0.9945
AM6	1.0048	0.9935
AM7	1.0056	0.9941

All our runs used our proposed DTW algorithm, described in section 1.2. The metric used is the normalized cross entropy cost (*Cnxe*) [11]. The results show almost no difference between the three types of features. Although PNCC should offer better accuracy in noisy conditions and the BUT systems had tokens for noise, this was not reflected in the results, with no big improvements obtained.

2.2 Official run results

The results obtained by the official runs on the evaluation database are shown in Table 5. We selected the first five best models, sorted after the *Actual Cnxe* metric. Because no tuning is made based on the development data set, the results on the evaluation data set are quite similar and the same conclusions can be drawn. Table 5 shows also the results per query type.

Table 5. Official 2015 runs

ID	Overall ACnxe	Type 1 ACnxe	Type 2 ACnxe	Type 3 ACnxe
AM6	1,0375	1,0384	1,0370	1,0376
AM7	1,0379	1,0391	1,0384	1,0365
AM1	1,0379	1,0383	1,0372	1,0385
AM4*	1,0380	1,0386	1,0378	1,0379
AM4	1,0381	1,0388	1,0378	1,0379

*PNCC model

It can be noticed that better results are obtained by query type 2, as these queries are longer, which may have affected the results. Posterior models (AM6/AM7) seem to offer minimal performance improvements, so we cannot draw a conclusion that posteriors are better suited for the STD task in difficult acoustical environments. Going further, we think improvements can be obtained by preprocessing the audio first, before extracting features, to remove noise or possible reverberation.

3. CONCLUSION

We have approached STD with a two-step process. A multilingual ASR is used as a phone recognizer for indexing the database, while a DTW based algorithm is used for searching a given query in the content database. We tested three types of features (MFCC, PNCC and Posteriorgrams) with two approaches to the phoneme recognizer (statistical HMMs and a neural STC approach). The results show no big improvement between each approach and feature types, in part because this year's database features very challenging acoustic environments, and our phonetizers return a lot of "noise" tokens or repeated phonemes, which reflected further upon our DTW algorithm.

4. ACKNOWLEDGEMENTS

The work has been funded by the Sectoral Operational Programme Human Resources Development 2007-2013 of the Ministry of European Funds through the Financial Agreements POSDRU/159/1.5/S/132395 and in part by the PN II Programme "Partnerships in priority areas" of MEN - UEFISCDI, through project no. 332/2014.

5. REFERENCES

- [1] I. Szöke, L.-J. Rodriguez-Fuentes, A. Buzo, X. Anguera, F. Metze, J. Proença, M. Lojka, and X. Xiong, "Query by example search on speech at MediaEval 2015," Working Notes Proceedings of the MediaEval 2015 Workshop, Sept. 14-15, 2015, Wurzen, Germany, 2015.
- [2] P. Schwarz, P. Matejka, J. Cernocky, *Towards Lower Error Rates in Phoneme Recognition*, in Proc. TSD2004, Brno, Czech Republic, 2004.
- [3] CMU Sphinx, An open source toolkit for speech recognition, Carnegie Mellon University, accessed august 2015, <http://cmusphinx.sourceforge.net/>
- [4] The International Phonetic Alphabet and the IPA Chart, <https://www.internationalphoneticassociation.org/>, accessed august 2015
- [5] J.S. Garofolo, et al., *TIMIT Acoustic-Phonetic Continuous Speech Corpus*, Linguistic Data Consortium, Philadelphia, 1993.
- [6] F. Kelly, N. Harte, *A comparison of auditory features for robust speech recognition*, In Proc. Of 18th European Signal Processing Conference (EUSIPCO-2010).
- [7] P. Schwarz, *Phoneme Recognition based on Long Temporal Context*, PhD Thesis, Brno University of Technology, 2009
- [8] Phoneme recognizer based on long temporal context, <http://speech.fit.vutbr.cz/software/phoneme-recognizer-based-long-temporal-context>, accessed august 2015.
- [9] Eastern European Speech Databases for Creation of Voice Driven Teleservices, <http://www.fee.vutbr.cz/SPEECHDAT-E/>, accessed august 2015.
- [10] P. Schwarz, P. Matejka, J. Cernocky, "Hierarchical Structures of Neural Networks for Phoneme Recognition", in Proc. ICASSP 2006, pp. 325-328, Toulouse, France, 2006.
- [11] L.-J. Rodriguez-Fuentes and M. Penagarikano. *MediaEval 2013 Spoken Web Search Task: System Performance Measures*. Technical report, GTTS, UPV/EHU, May 2013.
- [12] H. Andi Buzo, Horia Cucu, Iris Molnar, Bogdan Ionescu and Corneliu Burileanu, *Speed@MediaEval 2013: A Phone Recognition Approach to Spoken Term Detection*, in Proc. Mediaeval 2013 Workshop, Barcelona, Spain, 2013.