The ELiRF Query-by-Example STD systems for the Albayzin 2016 Search on Speech Evaluation

Sergio Laguna, Emilio Sanchis, Lluís-F. Hurtado, and Fernando García

Departament de Sistemes Informàtics i Computació
Universitat Politècnica de València, València, Spain
{slaguna,esanchis,lhurtado,fgarcia}@dsic.upv.es

Abstract. In this paper, we present two different systems to Query-by-Example Spoken Term Detection task. In both systems a first phase obtains a posteriorgram representation using a phoneme decoder. After that, the Subsequence DTW algorithm is performed to obtain the best matches between each query and the audio documents. Both systems differ in how the optimization process in the SDTW algorithm chooses the best path.

Keywords: Query-by-Example, Spoken Term Detection, Automatic Speech Recognition, Low Resources

1 Introduction

In this paper, we present two systems for the Query-by-Example Spoken Term Detection task. Both systems are based on a first phase where posterior phonetic probabilities for each frame are obtained. This phonetic probabilities are computed using the phoneme recognizer from Brno University of Technology [4] with a non-Spanish system, i.e. following a low resources strategy. After obtaining the phonetic probabilities, we apply a Subsequence Dynamic Time Warping algorithm [1, 2] to find the segments of the audio documents that best match the query utterance. The difference between both systems presented is how the minimization step in the SDTW algorithm selects the optimum path.

2 The ELiRF-SDTW QbE system

Our first system is based on the posterior phonetic probabilities computed with the BUT phoneme recognizer and the Subsequence Dynamic Time Warping algorithm using the cosine distance.

2.1 System Description

Preprocessing. We used the phoneme recognizer developed at the Brno University of Technology and tried the four available systems: Czech, English, Hungarian and Russian.

With this phoneme recognizer a vector representation of the audio files was built. For each frame, these decoders compute different number of features (45 for Czech, 39 for English, 61 for Hungarian and 52 for Russian), representing phonemic units. Each unit is composed of three states, so three posterior probabilities per phonetic unit and frame are computed. Three of the units do not represent actual phonemes, but they represent noise or silence. These posterior probabilities conform the feature vectors for each language, also called posteriorgrams [5].

In case of the Czech, Hungarian and Russian systems were trained on 8kHz audio, so the input audio files were downsampled from 16 kHz to 8 kHz (more information in section 2.2).

Subsequence Dynamic Time Warping. The search algorithm we used is based on Dynamic Programming (DP). In particular, we used the Subsequence Dynamic Time Warping, which is a variation of the well-known DTW algorithm. In our case, one of the sequences corresponded to feature vectors of one of the audio documents, and the other one represented a query. The SDTW algorithm is able to find multiple local alignments of the query within an audio document, by allowing it to start and end at any position of the audio document. Equation 1 shows the generic formulation of the SDTW:

$$M(i,j) = \begin{cases} +\infty & i < 0 \\ +\infty & j < 0 \\ 0 & j = 0 \\ \min\limits_{\forall (x,y) \in S} M(i-x, j-y) + D(A_i, B_j) & j \geq 1 \end{cases} \tag{1}$$

where M is the dynamic programming matrix; S is the set of allowed transitions, represented as pairs (x, y) of horizontal and vertical increments; A_i, B_j are the objects representing the i-th and j-th positions of their respective sequences; and D is a function that computes the distance or dissimilarity between two objects.

In this case, we do not use the usual transition set where the allowed movements are horizontal, vertical and diagonal. Instead, we modify the horizontal and vertical transitions so the paths found must have a length between half and twice the length of the query (see Figure 1).

Distance function. We tried different distance functions to obtain the dynamic programming matrix with the SDTW algorithm, like Kullback-Leibler divergence, cosine distance and inner product. After some experiments, we found the best results were obtained using the cosine distance:

$$cosine(u, v) = 1 - \frac{u \cdot v}{||u|| \cdot ||v||} \tag{2}$$

Filtering the detections. Once the SDTW algorithm had found the best alignments for each query utterance, the distance values were used to determine

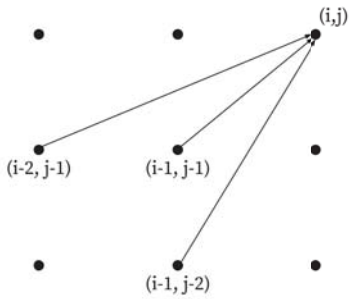

Fig. 1. Transitions set used in the SDTW algorithm.

a score for each detection. This score must indicate how likely it is that this detection is positive. For this reason, the score is inversely proportional to the distance. Finally, we select the best threshold to take the decision if a detection is positive or negative, so the system performance is maximized.

2.2 Train and development data

Train data. As we use the systems provided by the phoneme decoder from Brno University, the training data used is the following:

- The Czech system was trained on the Czech SpeechDat(E) database. This database contains about 12 hours of speech recorded over the Czech fixed telephone network in 8 kHz.
- The English system was trained on the TIMIT database. This database contains about 5 hours of read speech in 16 kHz.
- The Hungarian system was trained on the Hungarian SpeechDat(E) database. This database contains about 10 hours of speech recorded over the Hungarian fixed telephone network in 8 kHz.
- The Russian system was trained on the Russian SpeechDat(E) database. This database contains about 18 hours of speech recorded over the Russian fixed telephone network in 8 kHz.

Development data. This system was developed using the provided development dataset, which belongs to the Spanish MAVIR workshop material. This development dataset consists of 102 spoken queries and 2 audio documents amounting to about 1 hour of speech.

2.3 Preliminary results

Applying the approach presented, we evaluated the performance achieved by the system in the development data. The results obtained with the different systems

of the phoneme decoder are shown in Table 1. In this table is specified the best value achievable for the primary metric (Actual Term Weighted Value), also known as the Maximum Term Weighted Value (MTWV).

As we can check in Table 1, the best performance is achieved with the English recognizer.

Table 1. Results system 1

Recognizer	MTWV
Czech	0.0531
English	0.1991
Hungarian	0.0546
Russian	0.0681

3 The ELiRF-SDTW normalized QbE system

The second system developed is very similar to the previous one. In this case, we modified the minimization step of the Subsequence Dynamic Time Warping algorithm.

3.1 System description

In this new system the minimization of the SDTW algorithm is modified. Now, the search algorithm takes into account the length of the paths [3]. The equation 1 is modified as follows:

$$M(i,j) = \begin{cases} +\infty & i < 0 \\ +\infty & j < 0 \\ 0 & j = 0 \\ \min\limits_{\forall (x,y) \in S} \frac{M(i-x,j-y)+D(A_i,B_j)}{L(i-x,j-y)+1} & j \geq 1 \end{cases} \quad (3)$$

where $L(i,j)$ is the length of the best path ending in the point (i,j).

With this modification, if two paths have similar distance values but the length of their alignments are different, we use this information to select the best path.

3.2 Train and development data

As for the previous system, we use the Brno University phoneme recognizer to get the posterior probabilities. So, the train and development data are the same used for the first system. The information about this data is provided in section 2.2.

3.3 Preliminary results

The results obtained in the development set with the different systems of the phoneme decoder are shown in Table 2. The results are slightly better than the first system. In this case, the English recognizer also offers the best performance.

Table 2. Results system 2

Recognizer	MTWV
Czech	0.0555
English	0.2057
Hungarian	0.0848
Russian	0.0818

4 Final results

As we seen previously, both systems get similar results with the best phoneme decoder system. So, since the best performance is achieved with the ELiRF-SDTW normalized QbE system using the English recognizer, this is our primary system. The ELiRF-SDTW QbE system using the English recognizer is our contrastive system. In Table 3 are shown the results for both systems in development set.

Table 3. Final results in development set

System	MTWV
pri	0.2057
con1	0.1991

In Figure 2, we can see the Detection Error Tradeoff curve for the primary and contrastive systems with the development dataset.

5 Conclusions

In this work, we have presented two systems to Query-by-Example Spoken Term Detection task. The base of both systems is the Subsequence Dynamic Time Warping algorithm. The difference between these two systems lies in the optimization step of this algorithm. As both systems achieve similar results, we present them as primary and contrastive systems.

Acknowledgments This work was funded by the Spanish MEC under contract TIN2014-54288-C4-3-R.

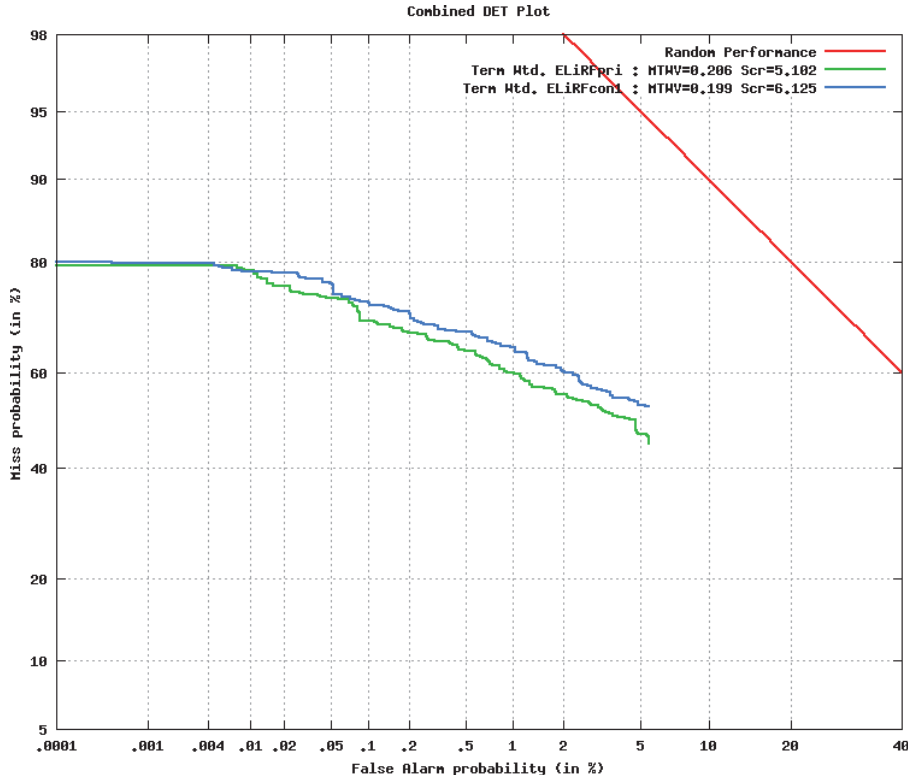

Fig. 2. DET curve for the development set.

References

1. Information Retrieval for Music and Motion, chap. Dynamic Time Warping, pp. 69–84. Springer Berlin Heidelberg, Berlin, Heidelberg (2007)
2. Anguera, X., Ferrarons, M.: Memory efficient subsequence DTW for Query-by-Example spoken term detection. In: 2013 IEEE International Conference on Multimedia and Expo. IEEE (2013)
3. Muscariello, A., Gravier, G., Bimbot, F.: Audio keyword extraction by unsupervised word discovery. In: INTERSPEECH 2009: 10th Annual Conference of the International Speech Communication Association (2009)
4. Schwarz, P.: Phoneme Recognition based on Long Temporal Context, PhD Thesis. Brno University of Technology (2009)
5. Zhang, Y., Glass, J.R.: Unsupervised spoken keyword spotting via segmental dtw on gaussian posteriorgrams. In: Automatic Speech Recognition & Understanding, 2009. ASRU 2009. IEEE Workshop on. pp. 398–403. IEEE (2009)