*INTERSPEECH 2016*
September 8–12, 2016, San Francisco, USA

# Segmented Dynamic Time Warping for Spoken Query-by-Example Search

*Jorge Proença, Fernando Perdigão*

Instituto de Telecomunicações and
Department of Electrical and Computer Engineering, University of Coimbra, Portugal
{jproenca,fp}@co.it.pt

## Abstract

This paper describes a low-resource approach to a Query-by-Example task, where spoken queries must be matched in a large dataset of spoken documents sometimes in complex or non-exact ways. Our approach tackles these complex match cases by using Dynamic Time Warping to obtain alternative paths that account for reordering of words, small extra content and small lexical variations. We also report certain advances on calibration and fusion of sub-systems that improve overall results, such as manipulating the score distribution per query and using an average posteriorgram distance matrix as an extra sub-system. Results are evaluated on the MediaEval task of Query-by-Example Search on Speech (QUESST). For this task, the language of the audio being searched is almost irrelevant, approaching the use case scenario to a language of very low resources. For that, we use as features the posterior probabilities obtained from five phonetic recognizers trained with five different languages.

**Index Terms**: Query-by-example, Spoken term detection, Dynamic Time Warping

## 1. Introduction

The task of searching large audio databases with a small query is commonly known as Spoken Term Detection (STD). It usually involves a text-based query and a spoken dataset of one language for which there are sufficient resources to build Automatic Speech Recognition (ASR) systems, resulting in a word-level indexing for the audio documents. Certain challenges have attracted research on the STD task, such as the NIST 2006 STD Evaluation [1] and the 2013 Open Keyword Spotting Evaluation [2].

The Query-by-Example (QbE) task differs from STD as the query must be speech based and no textual information is considered. Typically, there are no attempts to recognize word-level tokens, leading to a problem of finding audio using audio [3]–[6]. The need for QbE is found when the searched language is unknown or has few resources, or if multilingual databases are searched. The techniques for matching spoken queries to larger speech files usually involve the detection of unconstrained audio tokens in the data (zero-resources) [5] or the use of phonetic recognizers for other languages (low-resources) with the extraction of features such as posterior probabilities of phonemes [3], [4]. Most works use classical techniques such as Dynamic Time Warping (DTW) [3] or Acoustic Keyword Spotting (AKWS) [7].

The MediaEval task of Query by Example Search on Speech (QUESST) [8]–[10] proposes a suitable benchmark to evaluate QbE systems. With two editions – 2014 and 2015 – it distinguishes itself from other evaluations by introducing complex query-reference matches. These can be occurrences where a portion of the beginning or the end of the query may not match (small lexical variations), small extra content may be present between words on query or reference, or the searched words may appear in different order. The 2015's edition also added new relevant problems to tackle, as the audio can have different acoustic conditions with significant background or intermittent noise as well as reverberation, and there are queries that originate from spontaneous requests. These conditions further approach real case scenarios of a query search application, which is one of the underlying motivations of the challenge. The dataset is also multilingual and of mixed speaking styles, further increasing the challenging aspect of the task. A spoken document retrieval (SDR) solution is expected, as it is only necessary to retrieve the document that matches the query.

Systems for QbE search keep improving with recent advances such as combining spectral acoustic and temporal acoustic information [11]; combining a high number of subsystems using both AKWS and DTW and using bottleneck features of neural networks as input [12]; new distance normalization techniques [13] and several approaches to system fusion and calibration [14]. Some attempts have been made to address complex query types, by segmenting the query in some way such as using a moving window [15] or splitting the query in the middle [16]. Our approach is based on modifying the DTW algorithm to allow paths to be created in ways that conform to the complex types, which has shown success in improving overall results [17], [18].

For the presented methods, we reduce severe background noise by applying spectral subtraction, use five phonetic recognizers to extract posteriorgrams as features, improve and add modifications to DTW for complex query search (filler inside a query being the novelty), and implement improved fusion and calibration methods.

## 2. Dataset

The system described in this paper is evaluated with the QUESST 2015 dataset [9], which amounts to 18 hours of speech in 6 languages: Albanian, Romanian, Slovak, Czech, Portuguese and code switched Mandarin/English. 11662 recordings with an 8 kHz sampling rate were extracted from different sources of larger recordings such as broadcast news, lectures, read speech and conversations. The various languages are randomly distributed in the data, and no information is given to the participant about which language an utterance belongs to, requiring robust unsupervised approaches.

Queries were manually recorded in isolation by different speakers, in separate conditions from the utterances, emulating the use of a retrieval system with speech. Two sets of queries



http://dx.doi.org/10.21437/Interspeech.2016-1276

were created (445 for development and 447 for evaluation) and three types of queries were defined, exhibiting varying matching conditions to the utterances:

- Type 1 (T1): exact matches. The query should match the lexical form of incidences in the utterances without any filler content. For example, “brown elephant” as a query would match the utterance “The brown elephant is running”.
- Type 2 (T2): non-exact matches. Queries may have small lexical variations at the beginning or end compared to the occurrences in the search audio. An example would be the query “philosopher” matching an utterance containing “philosophy” (or vice-versa in this case). Queries with two or more words may have the words appear in a different order in the searched audio. Also, small irrelevant filler content in the utterances may be present (but not in the query). The matching possibilities of the query “brown elephant” in these cases are, for example, “elephant brown”, “elephant is brown”, “brown the elephant”.
- Type 3 (T3): conversational speech. Queries originate from spontaneous, more natural, requests, contrarily to T1 and T2 where queries are dictated. They may have the same non-exact match conditions as T2 queries, and may additionally present small filler content between words.

# 3. System Description

## 3.1. Noise filtering

First, we apply a high pass filter to the audio signals to remove low frequency artefacts. Then, to tackle the existence of substantial stationary background noise in both queries and reference audio, we apply spectral subtraction (SS) to noisy signals (not performed for high SNR signals, which worsened results). This implies a careful selection of samples of noise from an utterance. For this, we analyze the averaged log Energy of the signal, consider only values above -60dB, and determine high and low levels through median of quartiles as exemplified in Figure 1. Then, we calculate a threshold below which segments of more than 100ms are selected as “noise” samples, whose mean spectrum will be subtracted from the whole signal. Using dithering (white noise) to counterbalance the musical noise effect due to SS didn’t help. Nothing was specifically performed for reverberation or intermittent noise.

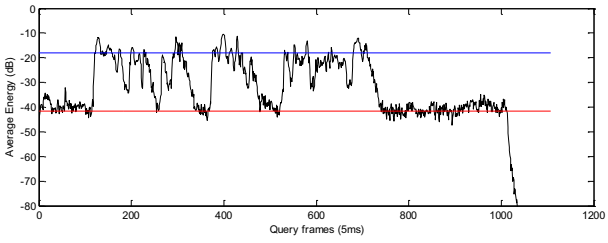


Figure 1: *Energy (dB) for one signal, with median of upper and lower quartiles as horizontal lines.*

## 3.2. Phonetic Recognizers

The next step is to run phonetic recognition on all audio and queries and extract frame-wise posterior probabilities of phonemes. An available external tool based on neural networks and long temporal context, the phoneme recognizers from Brno University of Technology (BUT) [19], is used. The three available systems for 8kHz audio, trained with SpeechDat-E databases [20], are employed: Czech (CZ), Hungarian (HU) and Russian (RU). Additionally, two new systems are trained with the same framework: English (EN - using TIMIT and Resource Management databases) and European Portuguese (PT - using annotated broadcast news data and a dataset of command words and sentences). Using different languages implies dealing with different sets of phonemes, and the fusion of the results will better describe the similarities between what is said in a query and the searched audio. This makes our system a low-resource one.

All de-noised queries and audio files were run through the 5 systems, extracting frame-wise state-level posterior probabilities (with 3 states per phoneme) to be analyzed separately. Figure 2 shows an example of the obtained posteriorgram for a clean query by using the Czech phonetic recognizer.

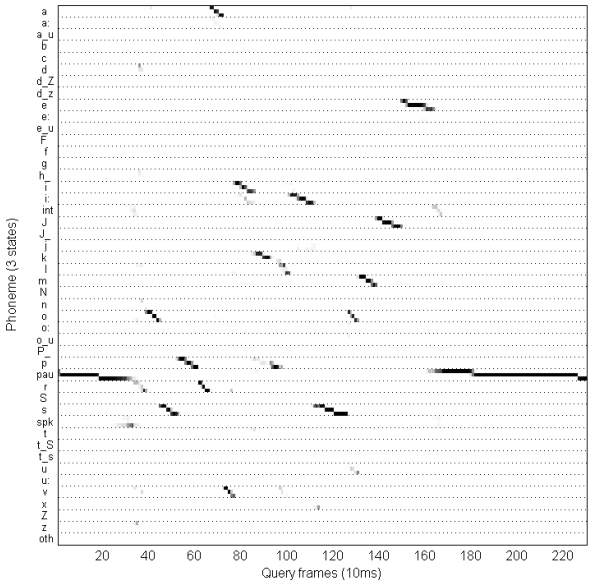


Figure 2: *State-level posterior probabilities for one query from the Czech recognizer.*

## 3.3. Voice Activity Detection

Silence or noise segments are undesirable for a query search, and were cut on queries from all frames that had a high probability of corresponding to silence or noise, if the sum of the 3 state posteriors of silence or noise phones is greater than a 50% threshold for the average of the 5 languages. To account for queries that may still have significant noise, this threshold is incrementally raised if the previous cut is too severe (the obtained query having less than 500ms).

## 3.4. Modified Dynamic Time Warping

Every query must then be searched on every reference audio. The posteriorgrams of a query and searched audio can be compared frame-wise with a local distance matrix where Dynamic Time Warping (DTW) can be applied. We implemented a version of the DTW approach for the proposed task, which will be modified in ways described next. The basis, as in [3], consists in obtaining the local distance from the dot product of posterior probability vectors of query and audio for

all frames obtaining a local distance matrix where DTW is applied. The path search may start at any location in the audio, and vertical, horizontal and diagonal jumps of unitary weight are allowed. The final path distance is normalized by the number of movements (mean distance of the path). This is the basic approach (named A1) and outputs the lowest normalized distance found (from the best path). It is the basis from which the following approaches will be constructed. In addition to separate searches on distance matrices from posteriorgrams of 5 languages, we add a 6th "language"/sub-system (called ML for multi-language) whose distance matrix is the average of the 5 matrices.

We employ several modifications to the DTW algorithm to allow intricate paths to be constructed that can correspond logically to the complex match scenarios of query and audio. It is not necessary to repeat full DTW for each, since backtrack matrices are saved where path modifications are explored. The modifications made are:

- (A2) Considering cuts at the end of query for lexical variations;
- (A3) Considering cuts at the beginning of the query;
- (A4) Allowing one horizontal 'jump' for situations where the audio may have filler content;
- (A5) Allowing word-reordering, where an initial part of the query may be found ahead of the last.
- (A6) Allowing one vertical 'jump' along the query, of maximum 33% of query duration (to address T3 where small fillers or hesitations in the query may exist).

Examples of A2 and A5 for clean audio can be seen in Figure 3 and Figure 4 . Further details of this implementation can be consulted in [18].

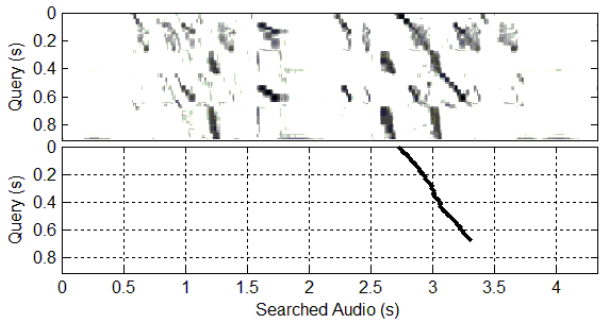


Figure 3: *Example of Query vs. Audio distance matrix (top) and the best path from A2 (bottom).*

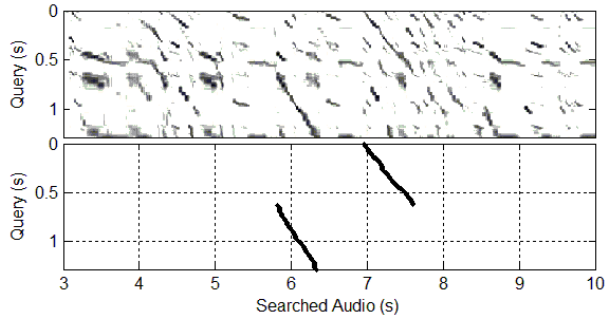


Figure 4: *Example of Query vs. Audio distance matrix (top) and the best path from A5 (bottom).*

### 3.5. Fusion and Calibration

The results of system performance will be presented by the scoring metrics of normalized cross entropy cost (Cnxe) and Actual Term Weighted Value (ATWV). Cnxe has been used for speaker/language recognition and evaluates system scores, with no concern for hard yes/no decision [21]. It interprets scores as log-likelihood ratios and measures the amount of information that is not provided by the scores compared to the ground truth where a perfect system would have Cnxe ≈ 0. ATWV evaluates system decision and takes into account false alarm and miss error rates. Both metrics consider a pre-defined false alarm error cost (Cfa=1) and a miss error cost (Cmiss=100), as well as a prior of the target trials (prior probability of finding a query in an audio file, Pt=0.0008).

At this stage, we have distance values for each audio-query pair for 6 sub-systems and 6 DTW strategies (36 vectors). First, modifications are performed on the distribution per query per strategy. While deciding on a maximum distance value to give to unsearched cases (such as an audio being too short for a long query), we found that drastically truncating large distances (lowering to the same value) improved both Cnxe and ATWV. Surprisingly, changing all upper distance values (larger than the mean) to the mean of the distribution was the overall best. We reason that since there are a lot of ground truth matches with very high distances (false negatives), lowering these values improves the Cnxe metric more than lowering the value of true negatives worsens it. The next step is to normalize per query by subtracting the new mean and dividing by the new standard deviation. Distances are transformed to figures-of-merit by taking the symmetrical value.

To fuse results of different strategies and languages we explore two separate approaches/systems, both using weighted linear fusion and transformation trained with the Bosaris toolkit [22], calibrating for the Cnxe metric by taking into account the prior defined by the task:

- Fusion of all approaches and all languages (36 vectors).
- Applying the Harmonic mean of the 6 strategies per language, obtaining 6 vectors (one per sub-system) and applying fusion. This is done to possibly prevent overfitting to the training data from weighing 36 vectors, and only languages are weighed.

From each fusion, final result vectors with only one value per audio-query pair are obtained for development and test data. To get a decision if a query is a match to the audio or not, a threshold is computed by finding the maximum TWV on the dev set, using the defined miss and false alarm costs and target prior.

Additionally, we provide side-info based on query and audio, added as extra vectors for all fusions. The 7 extra side-info vectors are: mean of distances per query before truncation and normalization from the best approach and language (the highest weighted from fusion of all); query size in frames and log of query size; 4 vectors of SNR values (original SNR of query and of audio, post spectral subtraction SNR of query and of audio).

## 4. Results

Four main systems are analyzed: fusion of all approaches and languages with and without side-info; fusion of harmonic mean with and without side-info. Table 1 summarizes the results of the Cnxe and ATWV metrics for the 4 systems.

| | Dev | | Eval | |
|---|---|---|---|---|
| Fusion Systems | Cnxe | ATWV | Cnxe | ATWV |
| All + side-info | **0.7782** | 0.2341 | 0.7866 | 0.2064 |
| Hmean + side-info | 0.7862 | 0.2195 | **0.7842** | 0.2017 |
| All, no side-info | 0.7873 | **0.2343** | 0.7930 | **0.2157** |
| Hmean, no side-info | 0.7957 | 0.2276 | 0.7915 | 0.2098 |

Table 1. *Results of Cnxe and ATWV for development and evaluation datasets using 4 fusion systems.*

Considering Cnxe as the main metric, it can be seen that the best result for the development set was the primary system that fused all languages and approaches plus some side-info. As suspected, the weighted combination of 36 vectors applied to the Eval set may be too over fitted to the Dev set, as the best Cnxe result on Eval was using the Hmean of approaches. The same does not hold true for the ATWV metric. The considered side-info always helped for Cnxe but leads to small decrease of ATWV.

Next, results are evaluated per language/sub-system and per DTW strategy, considering the Harmonic Mean method without side-info for the evaluation dataset. Figure 5 shows the Cnxe results by using each language individually, using the mean distance matrix ML, fusing the 5 languages (5l), or fusing the 5 languages and ML (All). The English system stands out as very poor performing. This may be due to being very sensitive to noisy conditions as the training audio for the phonetic recognizer is mostly very clean. Fusing the 5 languages at the end performs better than using the mean distance matrix ML, but fusing ML along with all languages provides the best results overall.

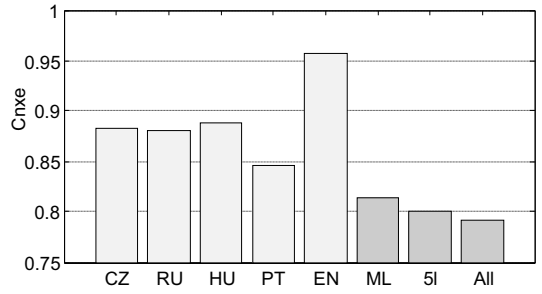


Figure 5: *Cnxe results per language sub-system and fusions for the Eval dataset.*

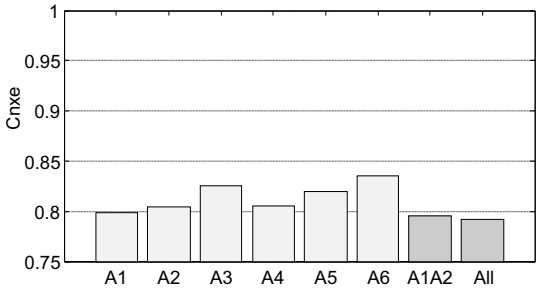


Figure 6: *Cnxe results per DTW approach and fusions for the Eval dataset.*

Analyzing the performance of each DTW strategy, as shown in Figure 6, it shows that A1 and A2 were the best ones and a fusion (harmonic mean) of only these 2 is also considered (A1A2 in the figure). Furthermore, A2 was the best performing one in the train set, and allowing these cuts at the end of the query may help in most cases due to co-articulation or intonation. The proposed strategy of allowing a jump in query (A6) performs badly and should be reviewed. Actually, a filler in a query may be an extension of an existing phone, which leads to a straight path and not a jump. Analyzing the best result on Eval per query type with side-info (All-0.7842, T1-0.7107, T2-0.8147, T3-0.8115), the exact matches of type 1 are the easiest to detect compared to other types.

Other improvements made through some steps of our system on the Dev set are also reported below (although the comparison may not be to the final approach). Using Spectral Subtraction resulted in 0.8130 Cnxe from 0.8368. Using per query truncation to the mean: 0.7873 Cnxe and 0.2343 ATWV, without truncation: 0.7939 Cnxe and 0.2256 ATWV.

Since a perfect Cnxe score would be 0, the obtained results above 0.77 may seem undesirable at first. It should be stated that the data with added noise and reverberation made the task extremely challenging. Although the spirit of QUESST discourages comparison between participating teams, it should be mentioned that the obtained results were the second best in 2015. When considering only the ground truth for audio and queries of low noise and no reverberation, even without recalibrating, the obtained results are more attractive: 0.576/0.542 Cnxe and 0.532/0.493 ATWV for dev/eval. Furthermore, applying the described methods to the data from QUESST 2014, excluding noise subtraction and side-info, the results of 0.4646 Cnxe and 0.5066 ATWV are very interesting as they would surpass that year's best.

## 5. Conclusions

Several steps were explored to tackle a Query by Example challenge, and the main contributions came from the following: a careful Spectral Subtraction to diminish background noise which greatly influences the output of phonetic recognizers; using the average distance matrix of all languages as a 6th sub-system for fusion; including side-info of query and audio; and per-query truncation of large distances. Including a DTW strategy that considers gaps in query did not prove very successful. This may be due to its target cases being too few in the dataset, and even some fillers in query being extensions and not unrelated hesitations. Using the harmonic mean of different DTW approaches instead of linearly fusing them leads to improved cross entropy costs on evaluation data.

Although the presented results mostly deal with very noisy data, the underlying conclusions should also be taken into account for cleaner data. Using bottleneck features could be an important step to improve our systems, and although we did not consider them yet, we focused on making improvements not related to the feature extractor. Manipulating the distribution of scores per query by drastically cutting values up to the mean indicates that more subtle normalization methods that alter the distribution should also be investigated.

## 6. Acknowledgements

This work was supported in part by Fundação para a Ciênica e Tecnologia under the project UID/EEA/50008/ 2013 (plurianual founding in the scope of the LETSREAD project). Jorge Proença is supported by the SFRH/BD/97204/2013 FCT Grant.

# 7. References


[1] J. G. Fiscus, J. Ajot, J. S. Garofolo, and G. Doddingtion, "Results of the 2006 spoken term detection evaluation," in *Proc. SIGIR*, 2007, vol. 7, pp. 51–57.

[2] NIST, "OpenKWS13 Keyword Search Evaluation Plan," Mar. 2013. http://www.nist.gov/itl/iad/mig/upload/OpenKWS13-EvalPlan.pdf

[3] T. J. Hazen, W. Shen, and C. White, "Query-by-example spoken term detection using phonetic posteriorgram templates," in *IEEE Workshop on Automatic Speech Recognition Understanding, 2009. ASRU 2009*, 2009, pp. 421–426.

[4] Y. Zhang and J. R. Glass, "Unsupervised spoken keyword spotting via segmental DTW on Gaussian posteriorgrams," in *IEEE Workshop on Automatic Speech Recognition Understanding, 2009. ASRU 2009*, 2009, pp. 398–403.

[5] C. Chan and L. Lee, "Model-Based Unsupervised Spoken Term Detection with Spoken Queries," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 7, pp. 1330–1342, Jul. 2013.

[6] F. Metze, X. Anguera, E. Barnard, M. Davel, and G. Gravier, "Language independent search in MediaEval's Spoken Web Search task," *Computer Speech & Language*, vol. 28, no. 5, pp. 1066–1082, Sep. 2014.

[7] I. Szöke, P. Schwarz, P. Matejka, L. Burget, M. Karafiát, M. Fapso, and J. Cernockỳ, "Comparison of keyword spotting approaches for informal continuous speech.," in *Interspeech*, 2005, pp. 633–636.

[8] X. Anguera, L. J. Rodríguez-Fuentes, A. Buzo, F. Metze, I. Szöke, and M. Peñagarikano, "QUESST2014: Evaluating Query-by-Example Speech Search in a zero-resource setting with real-life queries," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2015*, South Brisbane, Queensland, Australia, 2015, pp. 5833–5837.

[9] I. Szoke, L. Rodriguez-Fuentes, A. Buzo, X. Anguera, F. Metze, J. Proença, M. Lojka, and X. Xiong, "Query by Example Search on Speech at Mediaeval 2015," in *Working Notes Proceedings of the MediaEval 2015 Workshop*, Wurzen, Germany, 2015, vol. 1436.

[10] "The 2015 Query by Example Search on Speech (QUESST)," 2015. [Online]. Available: http://www.multimediaeval.org/mediaeval2015/quesst2015/. [Accessed: 21-Mar-2016].

[11] C. Gracia, X. Anguera, and X. Binefa, "Combining temporal and spectral information for Query-by-Example Spoken Term Detection," in *Proceedings of the 22nd European Signal Processing Conference (EUSIPCO)*, 2014, pp. 1487–1491.

[12] I. Szoke, L. Burget, F. Grezl, J. H. Cernocky, and L. Ondel, "Calibration and fusion of query-by-example systems—But SWS 2013," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 7849–7853.

[13] L. J. Rodriguez-Fuentes, A. Varona, M. Penagarikano, G. Bordel, and M. Diez, "High-performance Query-by-Example Spoken Term Detection on the SWS 2013 evaluation," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 7819–7823.

[14] A. Abad, L. J. Rodríguez-Fuentes, M. Penagarikano, A. Varona, and G. Bordel, "On the calibration and fusion of heterogeneous spoken term detection systems.," in *INTERSPEECH*, 2013, pp. 20–24.

[15] P. Yang, H. Xu, X. Xiao, L. Xie, C.-C. Leung, H. Chen, J. Yu, H. Lv, L. Wang, S. J. Leow, and others, "The NNI Query-by-Example System for MediaEval 2014," in *Working Notes Proceedings of the Mediaeval 2015 Workshop*, Barcelona, Spain, 2014.

[16] I. Szöke, M. Skácel, and L. Burget, "BUT QUESST 2014 system description," in *Working Notes Proceedings of the Mediaeval 2014 Workshop*, Barcelona, Spain, 2014, vol. 2014, pp. 1–2.

[17] J. Proença, A. Veiga, and F. Perdigão, "The SPL-IT Query by Example Search on Speech system for MediaEval 2014," in *Working Notes Proceedings of the Mediaeval 2014 Workshop*, Barcelona, Spain, 2014, vol. 1263.

[18] J. Proença, A. Veiga, and F. Perdigão, "Query by Example Search with Segmented Dynamic Time Warping for Non-Exact Spoken Queries," in *Proc 23rd European Signal Processing Conference (EUSIPCO)*, Nice, France, 2015, pp. 1691–1695.

[19] "Phoneme recognizer based on long temporal context, Brno University of Technology, FIT." [Online]. Available: http://speech.fit.vutbr.cz/software/phoneme-recognizer-based-long-temporal-context. [Accessed: 06-May-2015].

[20] P. Pollák, J. Boudy, K. Choukri, H. Van Den Heuvel, K. Vicsi, A. Virag, R. Siemund, W. Majewski, P. Staroniewicz, H. Tropf, and others, "SpeechDat (E)-Eastern European telephone speech databases," in *in the Proc. of XLDB 2000, Workshop on Very Large Telephone Speech Databases*, 2000.

[21] L. J. Rodriguez-Fuentes and M. Penagarikano, "MediaEval 2013 Spoken Web Search Task: System Performance Measures," Department of Electricity and Electronics, University of the Basque Country, TR-2013-1, 2013.

[22] N. Brummer and E. de Villiers, "The BOSARIS Toolkit User Guide: Theory, Algorithms and Code for Binary Classifer Score Processing," Technical report, 2011.