PRIVACY-PRESERVING QUERY-BY-EXAMPLE SPEECH SEARCH

José Portêlo[12], Alberto Abad[12], Bhiksha Raj[3], Isabel Trancoso[12]

[1] Instituto Superior Técnico, Universidade de Lisboa, Lisbon, Portugal ; [2] INESC-ID, Lisbon, Portugal
[3] Language Technologies Institute, Carnegie Mellon University, Pittsburgh PA, USA

ABSTRACT

This paper investigates a new *privacy-preserving* paradigm for the task of Query-by-Example Speech Search using Secure Binary Embeddings, a hashing method that converts vector data to bit strings through a combination of random projections followed by banded quantization. The proposed method allows performing spoken query search in an encrypted domain, by analyzing ciphered information computed from the original recordings. Unlike other hashing techniques, the embeddings allow the computation of the distance between vectors that are close enough, but are not perfect matches. This paper shows how these hashes can be combined with Dynamic Time Warping based on posterior derived features to perform secure speech search. Experiments performed on a sub-set of the SpeechDat Portuguese corpus showed that the proposed privacy-preserving system obtains similar results to its non-private counterpart.

***Index Terms*—** Query-by-Example Speech Search, Dynamic Time Warping, Secure Binary Embeddings, Data Privacy

1. INTRODUCTION

With the development of data recording and storage capabilities, it has become possible for many entities, both public and private, at the individual or corporate level, to store thousands of hours of speech audio for relatively low costs. Such speech collections may be composed of emergency calls or witnesses testimonies in the case of law enforcement agencies, business calls between service providers and their clients, doctors notes taken during medical appointments, examinations or surgeries, etc. Public entities and private companies usually store these recordings locally in their own servers, but an individual person might prefer to store them online, for convenience. These databases are extremely useful, as they allow for trend analysis and estimation, quality control and several other forms of service improvement. However, for these speech databases to be useful, it is necessary to know what is being said by which party.

A typical approach is to have a professional transcriber write down everything that was said or to run an automatic speech recognition (ASR) system over the database. A simpler, more robust and less expensive approach that is often employed is to simply search for specific terms or phrases that relate to specific topics of interest – inferences may be derived simply from the frequency of occurrence of these patterns. One version of this approach that is particularly relevant is the *Query-by-Example* Speech Search (QESS) approach, in which the keyphrase patterns to be searched for are specified through actual examples.

In all scenarios, however, privacy concerns arise. The data being mined frequently contain private data. The identity of the speaker is generally considered private and must not be exposed. Speakers also often reveal private information, such as account or social security numbers, demographic information, health information etc., all of which are clearly private and must not be accessible to anyone besides the person speaking it and the agency it is addressed to *in the context* it is intended to be heard in. Any access to this information outside of this setting is clearly unacceptable. In all scenarios, the identity of the people engaged in the conversations must be kept hidden, particularly the people who are not employed or similarly related to the entity storing the recordings.

Yet, when spoken recordings are mined, current technologies must expose all of the audio, including the private information in it, to the agency performing the mining, or even a hacker who may peer into the transactions. Ideally, this information must remain protected, even when allowing keyword search. This could be achieved, for instance, through a scheme that enables keyword search by analyzing *ciphered* information computed from the original recordings.

This, then, is the problem we address in this paper. We propose the combination of a QESS technique, namely the Dynamic Time Warping (DTW), with Secure Binary Embeddings (SBE) [1], to enable mining of encrypted data. SBE is a hashing technique that converts vector data to bit strings, through a combination of random projections followed by banded quantization. It possesses two interesting properties. First, the bit strings themselves represent encryptions with information theoretic guarantees of security, which prevent the recovery of the original data from the bit strings [1]. Second, it permits the computation of the Euclidean distance between vectors that are sufficiently close from the Hamming distance between the corresponding bit strings. The second property is an important characteristic of SBE hashes from our perspective, as this way the perfect-match restrictions inherent to other hashing techniques such as Locality-Sensitive Hashing [2] no longer apply. Therefore, they are much less dependent on the specific projections considered, enabling effective classification while retaining privacy.

The overall structure of this paper is as follows. In the next section we present an overview on Query-by-Example Speech Search techniques. Section 3 contains the Secure Binary Embeddings technique and its security guarantees. In Section 4 we describe our privacy-preserving query-by-example speech search scheme and we illustrate its performance with some experiments. Finally, in Section 5 we present some conclusions and directions for future work.

2. QUERY-BY-EXAMPLE SPEECH SEARCH (QESS)

One of the main approaches to mining large amounts of speech data is search for pre-specified keywords or phrases in it. In many situations, it is useful, or even necessary, to specify these key terms through *spoken* examples, rather than through other means of representation. In this “query-by-example” scenario, instances of these key terms in the test data to be mined are discovered through their acoustic match to the provided spoken examples. Query-by-Example Speech Search (QESS) is a particularly relevant problem for low-resource languages, and has recently gained significant re-

978-1-4673-6997-8/15/$31.00 ©2015 IEEE

search interest partially due to the success of the *Spoken Web Search* (SWS) task at the MediaEval evaluation series [3, 4]. In practice, the query-by-example task can be considered as a sort of generalization of the problem of speech search based on text queries. When the specific queries are known in advance of searching through a speech corpus, Keyword Spotting (KWS) approaches can be applied. On the other hand, when the data collection must be processed without prior knowledge of the queries, the so-called Spoken Term Detection (STD) task, a more elaborate procedure comprising a first indexing stage and a second search stage [5, 6] must be employed. The STD task has received considerable research attention in the recent past, partially due to the series of evaluations organized by NIST [7, 8, 9].

One common limitation of KWS and STD is that they are language-dependent. In both cases, conventional approaches rely somehow on a well-trained automatic speech recognition (ASR) system or on a set of phonetic models trained for the particular language of the speech collection. In well-resourced languages where such resources are available, even the query-by-example problem can be converted to KWS or STD: a simple straightforward approach would consist of an initial speech-to-text conversion of the query, followed by application of any of the methods used in text-query-based speech search. However, QESS is more appropriate in situations where specific acoustic or phonetic models may not be assumed for the language, such as in low-resource situations. It is also relevant in other situations, *e.g.* surveillance tasks or code-switched situations, where one may be looking for specific terms which are not well transcribed in the expected language of the corpus. In the particular case of QESS, some of the most recent approaches are based on template matching methods, such as different flavors of Dynamic Time Warping (DTW) of posterior derived features [10, 11]. As an alternative to the widespread template matching approaches, other systems use Acoustic Keyword Spotting (AKWS) [12, 13], exploiting in several ways acoustic models in multiple languages. A review of these and other methods can be found in [3, 4].

The baseline QESS systems considered in this work are based on DTW template-matching of posterior features provided by different language-dependent acoustic networks, similar to the ones described in [10, 11].

2.1. Feature extraction

Feature extraction is based on language-dependent phonetic networks that obtain posterior features exploiting multilayer perceptron (MLP) networks that are part of our in-house hybrid connectionist ASR systems for European Portuguese (*PT*), Brazilian Portuguese (*BR*), European Spanish (*ES*) and American English (*EN*). The phonetic class posterior probabilities are in fact the result of the combination of four MLP outputs trained with Perceptual Linear Prediction features (PLP, 13 static + first derivative), PLP with log-RelAtive SpecTrAl speech processing features (PLP-RASTA, 13 static + first derivative), Modulation SpectroGram features (MSG, 28 static) and Advanced Font-End from ETSI features (ETSI, 13 static + first and second derivatives). The language-dependent MLP networks were trained using different amounts of annotated data [14]. For the *PT* acoustic models, 57 hours of Broadcast News (BN) downsampled data and 58 hours of mixed fixed-telephone and mobile-telephone data were used. The *BR* models were trained with around 13 hours of BN downsampled data. The *ES* networks used 36 hours of BN downsampled data and 21 hours of fixed-telephone data. The *EN* system was trained with the HUB-4 96 and HUB-4 97 down-sampled data sets, that contain around 142 hours of TV and Radio Broadcast data. Each MLP network is characterized by the size of its input layer that depends on the particular parametrization and the frame context size (13 for PLP, PLP-RASTA and ETSI; 15 for MSG), the number of units of the two hidden layers (500), and the size of the output layer. In this case, only monophone units are modeled, resulting in MLP networks of 39 (38 phonemes + 1 silence) soft-max outputs in the case of *PT*, 40 for *BR* (39 phonemes + 1 silence), 30 for *ES* (29 phonemes + 1 silence) and 41 for *EN* (40 phonemes + 1 silence). Low-energy frames detected with the alignment generated by a simple bi-Gaussian model of the log energy distribution computed for each speech segment are removed. Finally, unit-length normalization is applied to the posteriors.

2.2. Dynamic Time Warping (DTW)

DTW is a time series alignment algorithm developed originally for speech recognition, which aims at aligning two sequences of feature vectors by warping the time axis iteratively until an optimal match between them is found. The two sequences can be arranged on the sides of a grid, one on the top and the other on the left hand side. Inside each cell a distance metric is computed, comparing the corresponding elements of the two sequences. For obtaining the best alignment between these two sequences, it is necessary to find a path through the grid that minimizes the total distance between them. This involves finding all possible paths through the grid and for each of them computing the overall accumulated distance, which corresponds to the minimum of the sum of the distances between the individual elements on the path divided by the sum of a weighting function. A common weighting function is the length of the path being analyzed. In order to force the optimal path to have an acceptable shape, several optimizations and constraints are usually used, such as monotonicity, continuity, maximum/minimum slope steepness, etc.

In our implementation of the DTW algorithm, we use a kernel that allows for the possible paths to advance only one cell at a time, either to the right, diagonally or down, thus enforcing monotonicity and continuity. Instead of using a sliding window approach for spotting a specific query in a long collection file sequence, we allow for the alignment between the query sequence and the long sequence to start and end at any arbitrary position of the long sequence. To do this, we impose some conditions to the computation of the best path length-normalized accumulated distances and we store all possible paths and the corresponding accumulated distance matrices. When the last cell is analyzed, we perform a backward step to find the minimum distance *crossing path* and the starting and ending frame of the query-term searched. More details of this process can be found in [11]. In this work, we produce a single score for each pair query-file corresponding to the negative length-normalized accumulated distance of the best *crossing path* match, given by

$$\text{Score}(KW, TS, p) = \min_{p \in \mathcal{P}} \sum_{p=1}^{P} \frac{d(KW, TS, p)}{\text{length}(p)} \qquad (1)$$

where KW are the frames corresponding to the target keyword, TS are frames corresponding to the target sentence, p is a possible path for aligning KW and TS, and $d(KW, TS, p)$ is the distance between KW and TS along path p. Thus, the speech search task here considered is spoken document retrieval, rather than query detection. Finally, we also apply a per-query zero-mean and unit-variance normalization (q-norm) to the detection scores.

3. SECURE BINARY EMBEDDINGS (SBE)

A *secure binary embedding* (SBE) is a scheme for converting real-valued vectors into bit sequences using band-quantized random projections. These bit sequences, which we will refer to as *hashes*, possess an interesting property: if the Euclidean distance between two vectors is lower than a threshold, then the Hamming distance between their corresponding hashes is proportional to the Euclidean distance between the vectors; if it is higher, then the hashes provide no information about the true distance between the two vectors. This scheme relies on the concept of Universal Quantization [15], which redefines scalar quantization by forcing the quantization function to have non-contiguous quantization regions.

Given an L-dimensional vector $\mathbf{x} \in \mathbb{R}^L$, the universal quantization process converts it to an M-bit binary sequence, where the m-th bit is given by

$$q_m(\mathbf{x}) = Q\left(\frac{\langle \mathbf{x}, \mathbf{a}_m \rangle + w_m}{\Delta}\right) \tag{2}$$

Here $\langle , \rangle$ represents a dot product. $\mathbf{a}_m \in \mathbb{R}^L$ is a projection vector comprising L i.i.d. samples drawn from $\mathcal{N}(\mu = 0, \sigma^2)$, Δ is a precision parameter, and w_m is a random dither drawn from a uniform distribution over $[0, \Delta]$. $Q(\cdot)$ is a quantization function given by $Q(x) = \lfloor x \bmod 2 \rfloor$. We can represent the complete quantization into M bits compactly in vector form:

$$\mathbf{q}(\mathbf{x}) = Q\left(\mathbf{\Delta}^{-1}(\mathbf{A}\mathbf{x} + \mathbf{w})\right) \tag{3}$$

where $\mathbf{q}(\mathbf{x})$ is an M-bit binary vector, which we will refer to as the *hash* of $\mathbf{x}$. $\mathbf{A} \in \mathbb{R}^{M \times L}$ is a matrix composed of the row vectors $\mathbf{a}_m$, $\mathbf{\Delta}$ is a diagonal matrix with entries Δ, and $\mathbf{w} \in \mathbb{R}^M$ is a vector composed from the dither values w_m.

The universal 1-bit quantizer of Equation 2 maps the real line onto $1/0$ in a banded manner, where each band is Δ_m wide. Figure 1 compares conventional scalar 1-bit quantization (left panel) with the equivalent universal 1-bit quantization (right panel).

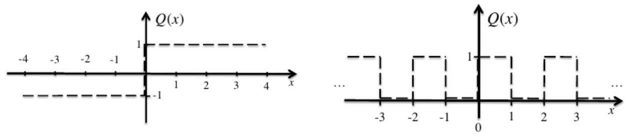

Fig. 1. 1-bit quantization functions.

The binary hash generated by the Universal Quantizer of Equation 3 has the following properties [1]: the probability that the i^{th} bits, $q_i(\mathbf{x})$ and $q_i(\mathbf{x}')$ respectively, of hashes of two vectors $\mathbf{x}$ and $\mathbf{x}'$ are identical depends only on the Euclidean distance $d = \|\mathbf{x} - \mathbf{x}'\|$ between the vectors and not on their actual values. As a consequence, the following relationship can be shown [1]: given any two vectors $\mathbf{x}$ and $\mathbf{x}'$ with a Euclidean distance d, with probability at most $e^{-2t^2 M}$ the normalized (per-bit) Hamming distance $d_H(\mathbf{q}(\mathbf{x}), \mathbf{q}(\mathbf{x}'))$ between the hashes of $\mathbf{x}$ and $\mathbf{x}'$ is bounded by:

$$\frac{1}{2} - \frac{1}{2}e^{-\left(\frac{\pi\sigma d}{\sqrt{2}\Delta}\right)^2} - t \le d_H(\mathbf{q}(\mathbf{x}), \mathbf{q}(\mathbf{x}')) \le \frac{1}{2} - \frac{4}{\pi^2}e^{-\left(\frac{\pi\sigma d}{\sqrt{2}\Delta}\right)^2} + t$$

where t is the control factor. The above bound means that the Hamming distance $d_H(\mathbf{q}(\mathbf{x}), \mathbf{q}(\mathbf{x}'))$ is correlated to the Euclidean distance d between the two vectors, if d is lower than a threshold (which depends on Δ). Specifically, for small d, $E[d_H(\mathbf{q}(\mathbf{x}), \mathbf{q}(\mathbf{x}'))]$, the expected Hamming distance, can be shown to be bounded from above by $\sqrt{2\pi^{-1}}\sigma\Delta^{-1}d$, which is linear in d. However, if the distance between $\mathbf{x}$ and $\mathbf{x}'$ is higher than this threshold, $d_H(\mathbf{q}(\mathbf{x}), \mathbf{q}(\mathbf{x}'))$ is bounded by $0.5 - 4\pi^{-2} exp\left(-0.5\pi^2\sigma^2\Delta^{-2}d^2\right)$, which rapidly converges to 0.5 and effectively gives us no information whatsoever about the true distance between $\mathbf{x}$ and $\mathbf{x}'$.

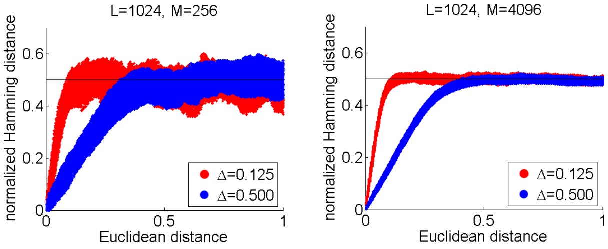

Fig. 2. SBE behavior as a function of Δ, for two values of M.

In order to illustrate how this scheme works, we randomly generated 1 million pairs of vectors in a high-dimensional space ($L = 1024$) and plotted the normalized Hamming distance between their hashes against the Euclidean distance between them (Figure 2). The number of bits in the hash is also shown in the figures. In all cases, once the normalized distance exceeds Δ, the Hamming distance between the hashes of two vectors ceases to provide any information about the true distance between the vectors. Changing the value of the precision parameter Δ allows us to adjust the distance threshold until which the Hamming distance is informative. Increasing the number of bits M leads to a reduction of the variance of the Hamming distance. A converse property of the embeddings is that for all $\mathbf{x}'$ except those that lie within a small radius of any $\mathbf{x}$, $d_H(\mathbf{q}(\mathbf{x}), \mathbf{q}(\mathbf{x}'))$ provides little information about how close $\mathbf{x}'$ is to $\mathbf{x}$. It can be shown that the embedding provides information theoretic security beyond this radius, if the embedding parameters $\mathbf{A}$ and $\mathbf{w}$ are unknown to the potential eavesdropper. Any algorithm attempting to recover a signal $\mathbf{x}$ from its embedding $\mathbf{q}(\mathbf{x})$ or to infer anything about the relationship between two signals sufficiently far apart using only their embeddings will fail to do so.

4. QUERY-BY-EXAMPLE SPEECH SEARCH USING SECURE BINARY EMBEDDINGS

The application of the SBE to a Query-by-Example Speech Search is straightforward: instead of evaluating the DTW algorithm using a distance metric appropriate for original arrays of features, the normalized Hamming distance between the corresponding SBE hashes is used, therefore hiding any relevant information from the party computing the algorithm.

The implementation of a *privacy-preserving* Query-by-Example Speech Search system is as follows: a party that possesses a collection of audio recordings which contain sensitive or otherwise private information extracts features from the audio, computes the corresponding SBE hashes, and stores them in a free, public-access datacenter. Later, when it is necessary to recover recordings containing a specific keyphrase, that party records one (or many) utterances of that keyphrase, extracts features from it, computes the corresponding SBE hashes with the same parameters $\mathbf{A}$, $\mathbf{w}$ and $\mathbf{\Delta}$ that were used before and performs the DTW algorithm over the entire collection. The only additional change is replacing the cosine or Euclidean distance with the normalized Hamming distance in Equation 1.

In this work, all the results are presented in terms of Maximum Term Weighted Value (MTWV) and DET curves, that are commonly used in NIST STD evaluations [7]. In this work we use a prior that approximately fits the empirical prior ($P_{target} = 0.01$) and two more suitable false alarm and miss error costs to our application scenario ($C_{fa} = 1$ and $C_{miss} = 100$). The corresponding β is 0.99.

ajuda	enviar	menu	saudação
alterar	fim	operador	seguinte
anterior	gravar	ouvir	stop
apagar	guardar	programar	telefonar
cancelar	ligar	rechamar	tocar
conferência	lista	repetir	transferir
continuar	marcar	sair	

Table 1. List of selected keywords.

	Posteriors				
	PT	*BR*	*ES*	*EN*	ALL
cosine	**0.812**	0.538	0.645	0.428	0.789
Euclidean	**0.800**	0.525	0.635	0.395	0.774

Table 2. QESS results, MTWV for baseline features.

4.1. Experiments using DTW

As a proof of concept, we ran experiments on a sub-set of the SpeechDat Portuguese corpus [16]. We selected 27 different keywords, shown in Table 1, from the SpeechDat application words list. The evaluation set are 481 utterances from one of the data categories of the Portuguese SpeechDat II corpora consisting of word spotting phrases using embedded application words. Most of the utterances contain only one application word (83.8%), but some utterances have zero (1.5%), two (12.0%) or three keywords (2.7%).

For each audio file, we extracted four sets of features: posteriors for European Portuguese (*PT*), Brazilian Portuguese (*BR*), European Spanish (*ES*) and American English (*EN*). The posteriors were extracted according to the procedure described in Section 2.1.

In our baseline experiments, without using SBE hashes, we evaluated all these sets of features separately but we also tried to perform a combination of all the posterior features for different languages at the distance matrices level, as this is known to improve results in some situations [17]. The results we obtained are presented in Table 2. Each row contains the results obtained when two different distance metrics are considered when performing the DTW algorithm. The cosine distance was used because it is the one that best adapts to the posterior features and the Euclidean distance was used because it is the one that best adapts to the SBE hashing scheme. Since all recordings contain speech from native European Portuguese speakers, it could be expected that the posterior-*PT* features provide the best results. The experiments considering all the posterior features at the same time did not improve the results obtained for the posterior-*PT* features, possibly because the results for the remaining language-dependent posteriors are rather worse in comparison.

4.2. Experiments using SBE hashes

There are two parameters that control the behavior of the Secure Binary Embeddings hashes: the quantization step size Δ and the number of bits M. The value of M by itself is not a useful number, as different values of L (dimensionality of the feature vectors) require different values of M; hence we report our results as a function of *bits per coefficient* (*bpc*), computed as M/L. The *bpc* allows us to govern the variance of the universal quantizer. *Leakage* in this context refers to the fraction of utterances containing the target keywords whose SBE hashes have a normalized Hamming distance below the threshold at which Hamming distance d_H is proportional to the Euclidean distance d with respect to any utterance in which we want to detect keywords. This threshold was empirically set at 0.475. The amount of leakage is exclusively controlled by Δ. Since the best baseline results were obtained for the posterior-*PT* features, we only analyzed them in experiments considering the SBE hashes. The results obtained are presented in Table 3.

leakage	$\sim 5\%$	$\sim 25\%$	$\sim 50\%$	$\sim 75\%$	$\sim 95\%$
bpc=2	–	–	0.776	0.776	0.782
bpc=4	–	0.731	0.798	0.790	0.787
bpc=8	–	0.784	0.744	0.794	0.784
bpc=16	–	**0.800**	0.799	0.798	0.797

Table 3. QESS results, MTWV for SBE hashes of posteriors-*PT*.

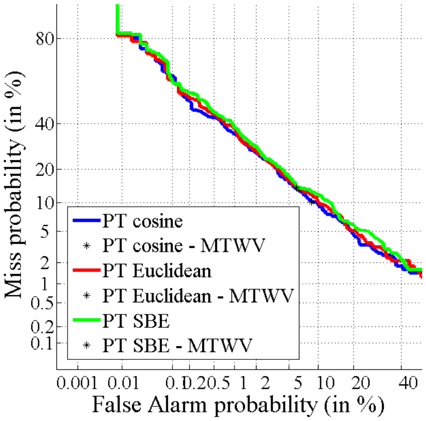

Fig. 3. DET curves for the baseline and best SBE hashes results.

As expected, increasing the value of Δ (and therefore increasing the amount of hashes that reveal information regarding the Euclidean distance between the original feature vectors) leads to better keyword detection results, specially when lower values of *bpc* are considered. Surprisingly, increasing the value of *bpc* does not lead to better keyword detection results. A possible reason for this may be that there is a clear separation between most of the utterances with and without a target keyword, which would mean that they are only slightly affected by the noise introduced when computing the SBE hashes. An interesting result is that the observed degradation when the SBE hashes are considered comes almost exclusively from the fact the SBE hashes relate to the Euclidean distance between the original vectors, and therefore they are very promising for obtaining privacy-preserving techniques for other speech processing tasks.

For the sake of completion, we present in Figure 3 the DET curves for the baselines using the cosine and Euclidean distances, as well as the best results using SBE hashes (*bpc*=16, 25% leakage).

5. CONCLUSIONS AND FUTURE WORK

The paper described a privacy-preserving query-by-example speech search system which yields similar results to the non-private counterpart. Our approach allows for searching audio databases for sentences containing specific keyphrases without the risks of disclosing personal or sensitive information. We are currently not only evaluating ways to extend the proposed work to use other forms of embeddings, but also analyzing mechanisms for more secure embeddings, as well as formally proving the non-invertibility of SBEs.

6. ACKNOWLEDGEMENTS

José Portêlo, Alberto Abad and Isabel Trancoso were supported by FCT grants SFRH/BD/71349/2010, PTDC/EIA-CCO/122542/2010 and UID/CEC/50021/2013.

7. REFERENCES

[1] P. Boufounos and S. Rane, “Secure Binary Embeddings for Privacy Preserving Nearest Neighbors”, in *Proc. Workshop on Information Forensics and Security (WIFS)*, Foz do Iguaçu, Brazil, December 2011.

[2] P. Indyk and R. Motwani, “Approximate Nearest Neighbors: Towards Removing the Curse of Dimensionality”, in *Proc. ACM Symposium on Theory of Computing*, Dallas, Texas, United States of America, 1998.

[3] F. Metze, X. Anguera, E. Barnard, M. Davel and G. Gravier, “The Spoken Web Search Task at MediaEval 2012”, in *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Vancouver, Canada, May 2013.

[4] X. Anguera, L.J. Rodriguez-Fuentes, I. Szoke, A. Buzo, F. Metze and M. Penagarikano, “Query-by-Example Spoken Term Detection Evaluation on Low-Resource Languages”, in *Proc. International Workshop on Spoken Language Technologies for Under-resourced Languages (SLTU)*, St. Petersburg, Russia, May 2014.

[5] J. Mamou, B. Ramabhadran and O. Siohan, “Vocabulary Independent Spoken Term Detection”, in *Proc. International ACM SIGIR Conference*, Amsterdam, Netherlands, July 2007.

[6] D. Miller, M. Kleber, C. L. Kao, O. Kimball, T. Colthurst, S. Lowe, R. M. Schwartz and H. Gish, “Rapid and Accurate Spoken Term Detection”, in *Proc. Interspeech*, Antwerp, Belgium, August 2007.

[7] J. G. Fiscus, J. Ajot, J. S. Garofolo and G. Doddington, “Results of the 2006 Spoken Term Detection Evaluation”, in *Proc. International ACM SIGIR Conference*, Amsterdam, Netherlands, July 2007.

[8] NIST, “OpenKWS13 Keyword Search Evaluation Plan”. Available at `http://www.nist.gov/itl/iad/mig/upload/OpenKWS13-EvalPlan.pdf`.

[9] NIST, “Draft KWS14 Keyword Search Evaluation Plan”. Available at `http://nist.gov/itl/iad/mig/upload/KWS14-evalplan-v11.pdf`.

[10] T. Hazen, W. Shen and C. White, “Query-by-Example Spoken Term Detection using Phonetic Posteriorgram Templates”, in *Proc. International Workshop on Automatic Speech Recognition & Understanding (ASRU)*, Merano, Italy, December 2009.

[11] 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 *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Florence, Italy, May 2014.

[12] I. Szöke, J. Tejedor, M. Fapso, and J. Colás, “BUT-HCTLab Approaches for Spoken Web Search”, in *Proc. MediaEval Workshop*, Pisa, Italy, September 2011.

[13] A. Abad and R. Astudillo, “The L^{2}F Spoken Web Search System for MediaEval 2012”, in *Proc. MediaEval Workshop*, Pisa, Italy, October 2012.

[14] A. Abad, J. Luque and I. Trancoso, “Parallel Transformation Network Features for Speaker Recognition”, in *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Prague, Czech Republic, May 2011.

[15] P. Boufounos, “Universal Rate-Efficient Scalar Quantization”, in *IEEE Trans. on Information Theory*, 58(3): 1861–1872, 2012.

[16] H. Hoge, H. S. Tropf, R. Winski, H. van den Heuvel, R. Haeb-Umbach, K. Choukri, “European Speech Databases for Telephone Applications” (EU-project LRE-633140), in *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Munich, Germany, April 1997.

[17] H. Wang, T. Lee, C.-C. Leung, B. Ma, and H. Li, “Using Parallel Tokenizers with DTW Matrix Combination for Low-Resource Spoken Term Detection”, in *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Vancouver, Canada, May 2013.