Efficient Query-by-Example Spoken Document Retrieval Combining Phone Multigram Representation and Dynamic Time Warping

Paula Lopez-Otero*, Javier Parapar, Alvaro Barreiro

Universidade da Coruña - CITIC, Facultad de Informática
Campus de Elviña S/N, 15071, A Coruña (Spain)

Abstract

Query-by-example spoken document retrieval (QbESDR) aims at finding those documents in a set that include a given spoken query. Current approaches are, in general, not valid for real-world applications, since they are mostly focused on being effective (i.e. reliably detecting in which documents the query is present) but practical implementations must also be efficient (i.e. the search must be performed in a limited time) in order to allow for a satisfactory user experience. In addition, systems usually search for exact matches of the query, which limits the number of relevant documents retrieved by the search. This paper proposes a representation of the documents and queries for QbESDR based on combining different-sized phone n-grams obtained from automatic transcriptions, namely phone multigram representation. Since phone transcriptions usually have errors, several hypotheses for the query transcriptions are combined in order to ease the impact of these errors. The proposed system stores the document in inverted indices, which leads to fast and efficient search. Different combinations of the phone multigram strategy with a state-of-art system based on pattern matching using dynamic time warping (DTW) are proposed: one consists in a two-stage system that intends to be as effective but more efficient than a DTW-based system, while the other aims at improving the performance achieved by these two

*This is to indicate the corresponding author.
Email address: {paula.lopez.otero,javierparapar,barreiro}@udc.gal (Paula Lopez-Otero*, Javier Parapar, Alvaro Barreiro)

systems by combining their output scores. Experiments performed on the MediaEval 2014 Query-by-Example Search on Speech (QUESST 2014) evaluation framework suggest that the phone multigram representation for QbESDR is a successful approach, and the assessed combinations with a DTW-based strategy lead to more efficient and effective QbESDR systems. In addition, the phone multigram approach succeeded in increasing the detection of non-exact matches of the queries.

Keywords: Query-by-example spoken document retrieval, Phone decoding, Phone n-grams, Phone posteriorgrams, Dynamic time warping

1. Introduction

The interaction with spoken contents has increased dramatically in the last few years due to the proliferation of audiovisual documents that are part of our daily life. This new paradigm of communication demands strategies for searching for contents of interest, creating the need for tools that allow the retrieval of spoken documents, task known as spoken document retrieval (SDR) [1]. SDR can be carried out using either written or spoken queries. This latter approach, known as query-by-example SDR (QbESDR), allows the communication with devices in a natural manner while easing the access to such technologies to visually impaired users.

The approaches for QbESDR found in the literature can be divided into two main groups: those based on automatic speech recognition (ASR), which imply transcribing both documents and queries into words or sub-words [2, 3, 4, 5, 6, 7]; and those that make use of pattern matching techniques, usually by finding alignments of the queries in the documents using the dynamic time warping (DTW) algorithm [8] or any of its variants [9, 10, 11, 12]. The main limitation of ASR-based strategies is the need for ASR resources in the language of interest, while pattern matching techniques are usually inefficient in terms of computational cost [11]. In addition, both strategies for QbESDR share an important limitation: they are intended to search for exact matches of the query.

This constraint does not recreate a real-world scenario, since a user might want to search for the exact query but also for lexical variations of it. In addition, when looking for queries with multiple terms, the documents that include the terms in a different order may be relevant for the search as well (for example, "president of Brazil" versus "Brazilian president").

Research in the field of QbESDR has been recently boosted by the organization of competitive evaluations such as Spoken Web Search [15, 16, 17] and Query by Example Search on Speech task [18, 19] at MediaEval; Spoken-Query&Doc Task at NTCIR [20, 21]; or query-by-example spoken term detection evaluation at Albayzín campaigns [22, 23, 24]. The zero resource speech challenge [13, 14] is devoted to unsupervised discovery of subword and word units from raw speech, which has QbESTD as one of its applications. The literature related to these evaluations shows a trend that consists in fusing the scores of the detections of different systems [25, 26, 27, 28, 29], which boosts the performance of the individual systems at the cost of increasing the computational demands of the search procedure. When considering a practical implementation for real-world scenarios, QbESDR approaches must be effective (i.e. they must be able to reliably detect in which documents the query is present) but also efficient (i.e. the search must be performed in a limited time) in order to allow for a satisfactory user experience. Hence, massive fusions can be effective but not efficient in practical terms, so new paradigms for QbESDR must be explored.

Two main contributions are presented in this paper, which aim at obtaining effective and efficient systems for real-world QbESDR applications:

- A novel approach for QbESDR based on phone n-gram representation, namely phone multigram representation, is proposed. Given a set of documents, their transcriptions are stored in inverted indices using different sizes of phone n-grams, i.e. the documents are stored tokenized in 1-grams, 2-grams and so forth. Afterwards, for each query, its equivalent tokenization in phone n-grams of different sizes is obtained in order to look for each term in the appropriate index, producing a score that indicates

how likely the given set of phone multigrams is present in each document. Additionally, in order to reduce the impact of transcription errors, several transcription hypotheses per query are obtained and searched, which leads to more reliable scores. This approach has several advantages:

  - A small amount of time is necessary for indexing and searching thanks to the efficiency of the inverted indexing and searching procedures.
  - Using phone multigrams for speech representation makes it possible to avoid taking into account the position where the match of each phone n-gram was found, since the smaller likelihood of matching long n-grams compensates that of matching short n-grams.
  - Since the order of the matching n-grams is not considered, the probability of finding non-exact matches of the queries increases.
  - This strategy can be used in a cross-lingual manner, since the language of the phone decoder used to obtain phone transcriptions does not necessarily have to match the language spoken in the documents and queries.

  This approach is inspired by [30, 31], where a similar strategy was used for text retrieval in noisy documents obtained by optical character recognition (OCR). The application scenario is very similar, since both OCR and phone transcriptions have errors that do not allow the search for exact matches of a query.

- Two different combinations of the phone multigram approach with a strategy based on DTW are presented:

  - The first combination consists in a two-stage system: first, the phone multigram system is used to look for candidate matches of the queries in the documents; then, these matches are re-scored using a DTW-based strategy in order to decide whether to keep them or discard them. This strategy increases the efficiency of the search process,

since the number of query-document pairs that have to be evaluated with DTW (which is significantly more costly than the phone multigram strategy) is reduced to a great extent.

– A second combination is proposed, which consists in fusing the output scores of the phone multigram and DTW-based systems. Given that the phone multigram approach is computationally efficient, running these two systems hardly affects the efficiency of the search. Moreover, this combination leads to a relevant improvement in terms of effectiveness as a result of combining different pieces of evidence produced by heterogeneous systems.

The rest of this paper is organized as follows: Section 2 describes the related work; Section 3 presents the phone multigram approach for QbESDR; Section 4 overviews the DTW-based system used in this work; Section 5 presents two techniques for combining the two aforementioned approaches; Section 6 describes the experimental framework; experimental results and a discussion are presented in Section 7; Section 8 reviews other results reported in the literature for the experimental framework used in Section 7; lastly, conclusions and future work are summarized in Section 9.

2. Related work

QbESDR techniques based on ASR usually inherit the methodology employed in SDR using written queries [32, 33, 34]. Nevertheless, only documents must be transcribed when doing SDR with written queries but, in the case of QbESDR, both queries and documents must be converted into a textual representation [4], which increases the noise on the data to be processed since errors can be present in both documents and queries. Indeed, as suggested by the results shown in [23], errors on query transcriptions lead to degraded search performance when comparing spoken and written queries. In addition, the performance of such approaches is reasonable in controlled scenarios where the word error rate is reduced, and it relies on the availability of an ASR system for the

language of interest. In the absence of an ASR system for a given language or for out-of-vocabulary SDR, the employment of sub-word transcriptions is common, as well as the use of cross-lingual approaches, i.e. using an ASR system in a different language to achieve a transcription of the documents and queries into phones or sub-words such as syllables [2, 3] or n-grams [35, 5, 6, 7]. The use of n-gram representation is very common in tasks dealing with noisy contents such as text retrieval from OCR data [30, 31], cross-language information retrieval from misspelled queries [36] or language identification on noisy texts [37]; and also in other tasks such as data extraction from web pages [38], author profiling [39] or plagiarism detection [40]. To conclude the overview of approaches based on ASR strategies, the recent popularity of ASR systems featuring multilingual representations for ASR and keyword search on low-resource languages must be highlighted [41, 42].

QbESDR approaches based on pattern matching techniques consist in representing the spoken documents and queries using frame-level vectors and employing this representation to search for an alignment between query-document pairs by means of the DTW algorithm [8] or any of its variants [9, 10, 11, 12]. This technique allows the use of cross-lingual strategies with acceptable results using low-resource approaches. In such techniques, speech representation usually relies on Gaussian posteriorgrams [43], where speech frames are represented by the posterior probabilities of each Gaussian in a Gaussian mixture model [10, 44, 26, 45]; or on phone posteriorgrams, which consist in time vs. class matrices representing the posterior probability of each phone class for each instant of time, and they can be obtained using phone decoders that are not necessarily developed in a given target language [46, 47, 48, 29, 49]. Zero-resource QbESDR approaches have also become very popular due to their reduced amount of required resources; in this scenario, documents and queries are usually represented by features straightforwardly extracted from the waveforms such as Mel frequency cepstral coefficients (MFCCs) [50, 51, 4], perceptual linear prediction coefficients [52], short-time frequency domain linear prediction features [53], or large sets of features followed by feature selection [54]. The main disadvantage

of all these strategies based on pattern matching is the time required for searching, although some efficient variants of DTW have been proposed in order to cope with this issue [11]. In addition, DTW aims at searching for exact matches of the query, which complicates the search of lexical variations of the queries or word reorderings in queries with multiple terms. Some approaches have been presented to overcome this issue, such as [55, 28, 56]. In these works, strategies consisting in modifying some constraints of the DTW algorithm were proposed, such as considering cuts at the beginning and/or the end of the query, allowing a horizontal jump to cope with filler content between the different words of the query, or looking for the last part of the query before the initial part to deal with word reorderings. The results presented in [28, 55] show that combining these variations of DTW yields good QbESDR results, but this is achieved at the expense of increasing the search time to a great extent. In addition, as reported in [56], the top-performing strategies are classic DTW and the approach that allows a cut at the end of a query, while the specific approaches for word reordering did not yield good individual results.

There are few works in the literature focusing on increasing the efficiency of QbESDR strategies. In [49], a system based on bag of acoustic words (BoAW) representation [57] was used to obtain potential candidate matches of the queries within the documents. Afterwards, those candidates were validated using a DTW-based approach, leading to variable results dependent on the decision threshold used for candidate selection.

3. Proposed method: phone multigram representation for QbESDR

The search on speech approach presented in this paper consists in representing the documents by means of phone multigrams, i.e. a combination of different-sized phone n-grams, and their subsequent storage and search in inverted indices. The proposed representation accounts for transcription errors and allows the fast search of queries, even with lexical variations and word reorderings, in large collections of spoken documents. This approach consists of

two stages: indexing and search. The first step encompasses the process of transcribing the documents and creating a searchable index. The second stage consists in obtaining transcriptions of the queries, searching for them in the index and generating scores that indicate how likely each query was found in each document. The rest of this section presents the proposed system, preceded by a brief description of the speech transcription strategy, a necessary step for obtaining phone multigrams, as well as an overview of the implementation of QbESDR using inverted indices.

3.1. Speech transcription

The transcription of spoken queries and documents can be done by means of phone decoding of the audio signals. Phone decoding consists in converting a speech utterance into a textual representation where each term represents a phone (i.e. a sound). This procedure is usually carried out by means of an ASR system whose language model is a phone loop, i.e. there are no phonotactic constraints, all the transitions from one phone to any of the others are possible [58]. Hence, given a speech utterance and a phone decoder with n_U phone units $U = \{u_1, \ldots, u_{n_U}\}$, the phone decoder generates a phone lattice, i.e. a directed acyclic graph with a single start point and edges labeled with a phone hypothesis and a likelihood value [59]. Phone lattices allow the extraction of the 1-best transcription (i.e. the most likely phone transcription of the speech utterance) but also other less likely transcriptions, namely n-best transcriptions. Since the error rates obtained with phone decoding can be very high, especially in unconstrained data, using only the 1-best transcription might not be accurate enough for performing QbESDR. Therefore, it is possible to extract several hypotheses for the phone transcription of a speech utterance and use them together in order to mitigate the transcription errors. Hence, it is possible to perform QbESDR on documents represented by lattices or n-best transcriptions ($n \geq 1$). In practical terms, lattices include much more information than n-best transcriptions, but using them to represent the documents is considerably slower than using n-best transcriptions, since the number of possible paths is

dramatically reduced in the latter alternative [5].

3.2. QbESDR using inverted indices

Before describing the approaches presented in this paper, an introduction to the indexing and search procedure must be done. Inverted indices are commonly used for text information retrieval, since this data structure allows fast an efficient search while achieving an optimal use of storage space [60]. Given a set of n_Ω documents $\Omega = \{D_1, \ldots, D_{n_\Omega}\}$ to be indexed, each document D_i is represented by a set of n_{D_i} terms $D_i = \{t_1, \ldots, t_{n_{D_i}}\}$. The inverted index stores, for each term, all the documents that include that term. This is faster than storing each document along with its corresponding terms, since searching a term would imply going through all the documents looking for the term of interest. Depending on the tokenization procedure and on the specific needs of the application, the considered terms can be words, n-grams or graphemes, to cite some examples. In the context of QbESDR, since speech utterances are converted to sequences of phones, terms can be either phones or phone n-grams created by joining adjacent phones into a single term (it must be noted that phones are equivalent to phone 1-grams).

Once indexing is done, a spoken query can be searched within the inverted index formulating a search query. Formally, given a spoken query Q whose transcription is $Q = \{u_1, \ldots, u_{n_Q}\}$, where u_i represents a phone unit, it can be transformed into a search query $Q^s = \{t_1, \ldots, t_{n_Q^s}\}$, where t_i represents a query term. Note that the number of terms in the search query n_Q^s depends on the tokenization used: $n_Q^s = n_Q$ when dealing with phone 1-grams (namely phones), but this is not true when dealing with phone n-grams when $n \neq 1$. In any case, the tokenization used to formulate the query must comprise n-gram sizes that are present in the index.

In QbESDR, a score must be assigned to each query-document pair in order to indicate how likely the query matches each document, since it must be decided whether the query is present in the document or not. This is similar to the information retrieval scenario, where a relevance score must be assigned to the

retrieved documents. Hence, it is straightforward to borrow scoring models from the information retrieval literature, such as the widely used vector space model (VSM) [61]. This model aims at representing the documents and queries by means of vectors, which can be straightforwardly compared by computing their dot product. These vectors are usually obtained using the tf-idf weighting scheme, which considers two types of weights:

- Term frequency $tf(t, D)$ of term t in document D. This measure assigns a weight to each term in a document that depends on the number of occurrences of the term in the document [62]. The motivation behind this measure is that when a term appears many times in a document, this document is probably relevant for a search of that term.

- Inverse document frequency $idf(t)$ of term t. This measure gives more weight to those terms that are least frequent in the set of documents, since they are considered to be more relevant [62].

In this work, a scoring function based on the $tf \cdot idf$ VSM implementation of Lucene[1] was used. The score of a query Q and a document D is computed as:

$$score(Q, D) = coord(Q, D) \frac{\mathbf{V}(Q) \cdot \mathbf{V}(D)}{|\mathbf{V}(D)|} \tag{1}$$

where $\mathbf{V}(D)$ and $\mathbf{V}(Q)$ are the vectors of the document and the query, respectively; $|\mathbf{V}(D)|$ is the length-normalization factor for document D; and $coord(Q, D)$ is the coordination factor, which measures the number of terms in query Q that are present in document D, since the document will be considered to be more relevant to the search if many of the query terms are found in the document.

In practical terms, Eq. (1) is implemented in Lucene as:

$$score(Q, D) = \frac{coord(Q, D)}{n_D} \sum_{t \in Q} \left(tf(t, D) idf(t)^2 \right) \tag{2}$$

[1] http://lucene.apache.org/

where n_D is the aforementioned document length-normalization factor, which is equal to the number of terms in document D. In addition, $tf(t, D)$ is computed as

$$tf(t, D) = \sqrt{frequency(t, D)} \tag{3}$$

where $frequency(t, D)$ is the number of occurrences of t in document D; $idf(t)$ is computed as

$$idf(t) = 1 + log\left(\frac{n_\Omega + 1}{docFreq(t) + 1}\right) \tag{4}$$

where n_Ω is the total number of indexed documents and $docFreq(t)$ is the number of documents with occurrences of term t; and $coord(Q, D)$ is computed as

$$coord(Q, D) = \frac{occurrences(Q^s, D)}{n_Q^s} \tag{5}$$

where $occurrences(Q^s, D)$ is the number of terms of the search query Q^s that are present in document D, and n_Q^s is the total number of terms of the search query.

3.3. Phone multigram representation for QbESDR

In this paper, an approach inspired by previous work in text retrieval on noisy documents obtained by OCR is proposed [30, 31]. This strategy consists in, instead of using a single tokenizer for spoken document representation, combining different tokenizers with several objectives: (1) ease the impact of transcription errors on the results: (2) allow fast search within inverted indices without having to take positional information into account; (3) improve the non-exact matching of queries, which is an unsolved problem using the most common strategies for QbESDR [55, 28, 56, 5, 6].

Given a set of documents, first their 1-best transcriptions are obtained using a phone decoder and then these transcriptions are subsequently indexed. Instead of creating a single index for phone n-grams (for a fixed value of n), the proposed strategy consists in storing several n-gram indices for different values of $n \in \{min_{ngram}, \ldots, max_{ngram}\}$. Figure 1 presents an example of the tokenization and indexing procedure: given a spoken document to index, first its phone

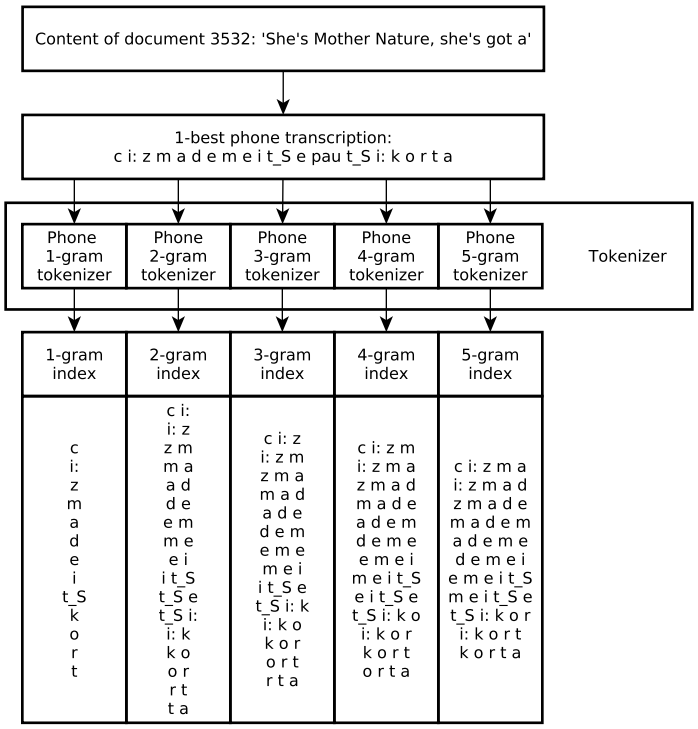

Figure 1: Example of the indexing approach proposed in this paper: the 1-best phone transcription of a spoken document is obtained and then it is tokenized and indexed. It must be noted that, in this example, the tokenizer is composed of five phone n-gram tokenizers with $n = 1, \ldots, 5$; this leads to five indices, one for each n-gram tokenizer.

transcription is obtained using a decoder, and then it is tokenized and indexed using different n-gram sizes from $min_{ngram} = 1$ to $max_{ngram} = 5$.

Since the documents are stored using different tokenizations, the search queries must be constructed in an equivalent manner. Then, for each spoken query, its 1-best transcription is obtained and tokenized in the same way as in the indexing procedure. Then, a search query is created such that it implies searching for each n-gram found in the query in its corresponding index (i.e. 1-grams are searched within the 1-gram index and so forth). This strategy will produce a score for each index following Eq. (2), but all these scores must be

combined in order to obtain a single one per query. This is done as follows:

$$score(Q, D) = \sum_{i=min_{ngram}}^{max_{ngram}} score(Q, D, i) \tag{6}$$

where $score(Q, D, i)$ is the score corresponding to the i-gram index computed following Eq. (2).

Provided that the transcriptions of queries and documents usually have a high phone error rate, a strategy was designed to reduce the impact of these errors, which consists in extracting multiple transcription hypotheses for each spoken query and combine the search results of all of them. Given the n-best transcriptions of a query, the first n_{hyp} hypotheses $\{Q^1, \ldots, Q^{n_{hyp}}\}$ are tokenized and searched within the index, and their scores are combined into a single score per document as follows:

$$score(Q, D) = \max_{i \in 1,\ldots,n_{hyp}} score(Q^i, D) \tag{7}$$

where Q^i is the i^{th} transcription of query Q. This strategy for query combination is more suitable than averaging the scores of all the query hypotheses since it is equivalent to searching the query transcription hypothesis that yields the best (highest) score. This procedure is depicted in Figure 2.

As mentioned above, $score(Q, D)$ is used to decide whether query Q is present in document D or not, so it is necessary to establish a decision threshold that is valid for all documents and queries. Eq. (2) includes a document length-normalization factor that aims at making the scores of different documents comparable. A common technique for query normalization in QbESDR is applied in this system in order to make the scores of different queries comparable, since it is known that the scores corresponding to different queries usually follow different distributions [63]. Specifically, the z-norm [63] was applied, which is widely used in this scenario: given a set of n_m documents $D_Q = \{D^1, \ldots, D^{n_m}\}$ that matched query Q, their scores are normalized as follows:

$$score_{z-norm}(Q, D^i) = \frac{score(Q, D^i) - \mu_Q}{\sigma_Q} \tag{8}$$

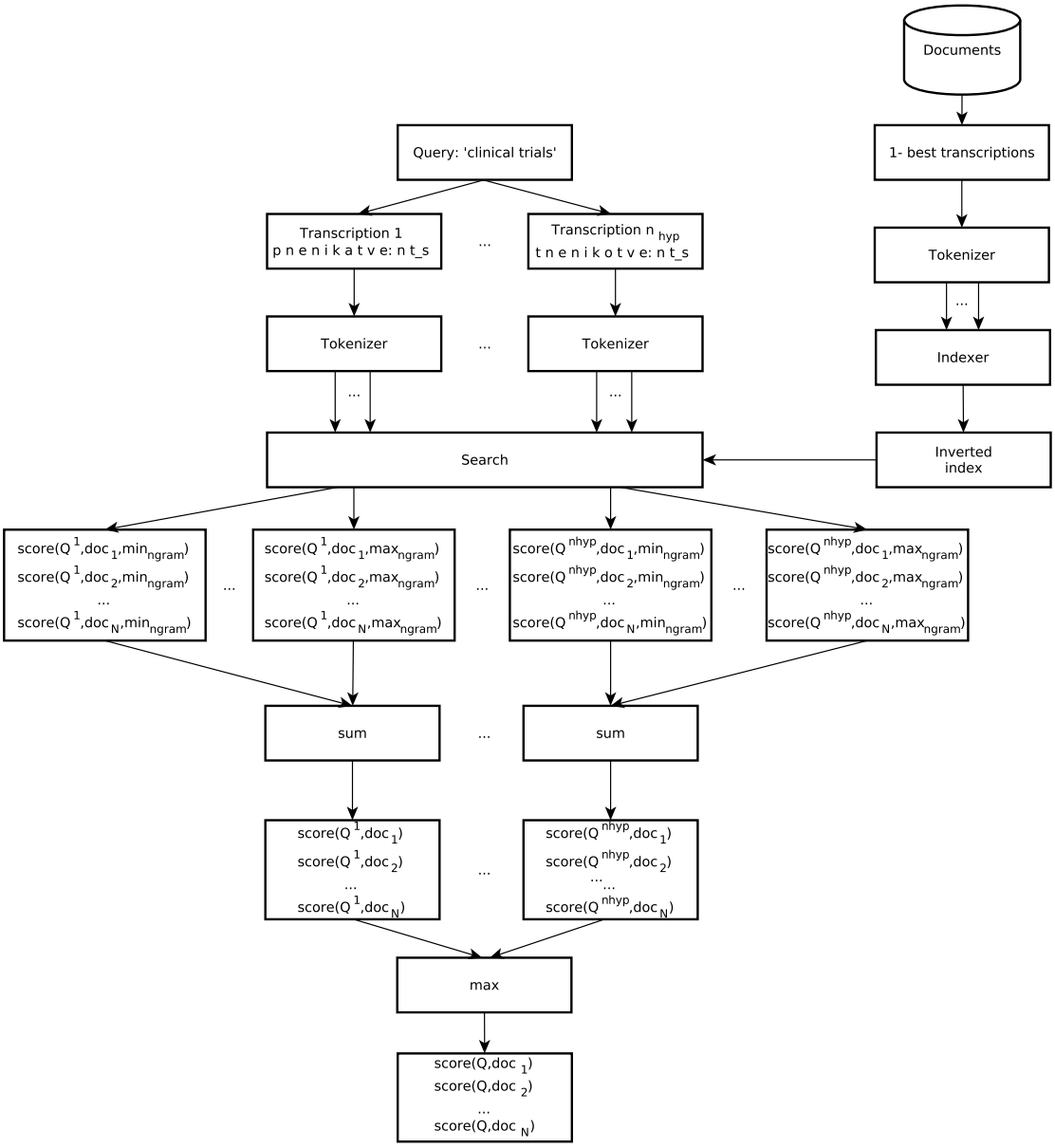

Figure 2: Block diagram of the indexing and search strategy. The figure shows in detail how the scores obtained from the different transcription hypotheses are combined into a single score per query/document pair, as described in Eq. (7). As in Figure 1, the tokenizer is composed of several phone n-gram tokenizers.

where

$$\mu_Q = \frac{1}{n_m} \sum_{i=1}^{n_m} score(Q, D^i) \tag{9}$$

is the mean of the scores of D_Q and

$$\sigma_Q = \sqrt{\frac{1}{n_m - 1} \sum_{i=1}^{n_m} |score(Q, D^i) - \mu_Q|^2} \tag{10}$$

is the standard deviation of the scores of D_Q. In this way, all the scores have a distribution with zero mean and unit variance, which makes it possible to establish the same decision threshold regardless of the query.

4. QbESDR using dynamic time warping

DTW is widely used in pattern matching-based approaches for QbESDR. This type of systems usually consist of three stages: feature extraction, search and score normalization. First, frame-level features are extracted from the waveforms of both queries and documents. Then, in the search stage, each query is matched against every spoken document in order to obtain scores that indicate how likely the query was found in the document. Lastly, these scores are normalized in order to make them comparable among different documents and queries. The rest of this section describes the system used in this work in detail.

4.1. Feature extraction

As mentioned in Section 2, different features are used in DTW-based systems for search on speech. Among all of them, phone posteriorgrams are widely used for QbESDR due to their acceptable results in cross-lingual scenarios: it is possible to extract phone posteriorgrams of spoken documents in a given language using a phone decoder trained for a completely different language and still obtain a valid representation of the spoken contents [64, 65, 66, 67, 48, 29]. Hence, in this work, phone posteriorgrams were used for query and document representation. Given a spoken document and a phone decoder with U phone units, the posterior probability of each phone unit is computed for each time

frame, leading to a set of vectors of dimension U that represents the *a posteriori* probability of each phone unit at every instant of time. After obtaining the posteriors, a Gaussian softening is applied in order to have Gaussian distributed probabilities [68].

4.2. Search algorithm

Let $Q = \{\mathbf{q}_1, \ldots, \mathbf{q}_{n_Q}\}$ and $D = \{\mathbf{d}_1, \ldots, \mathbf{d}_{n_D}\}$ be the phone posteriorgrams of a query and a document with n_Q and n_D frames, where $\mathbf{q}_i$ and $\mathbf{d}_j$ are feature vectors of dimension U and $n_Q \ll n_D$. DTW aims at finding the best alignment path between Q and D. There are several variants of DTW that can be used for search on speech [8, 9, 10, 11, 12]. In this work, subsequence DTW (S-DTW) was used [9], since it allows alignments between a short sequence (the query) and a longer sequence (the document); in other words, the query does not have to match the whole document but only a part of it.

First, a cost matrix $\mathbf{M} \in \Re^{n_Q \times n_D}$ is defined, where the rows and columns correspond to the frames of the query and the document, respectively. Each element $M_{i,j}$ of the cost matrix represents the cost corresponding to frame $\mathbf{q}_i$ in the query and frame $\mathbf{d}_j$ in the document, which is defined as

$$M_{i,j} = \begin{cases} c(\mathbf{q}_i, \mathbf{d}_j) & \text{if} \quad i = 1 \\ c(\mathbf{q}_i, \mathbf{d}_j) + M_{i-1,0} & \text{if} \quad i > 1,\ j = 1 \\ c(\mathbf{q}_i, \mathbf{d}_j) + M^*(i,j) & \text{otherwise} \end{cases} \tag{11}$$

where $c(\mathbf{q}_i, \mathbf{d}_j)$ is a function that defines the cost between query vector $\mathbf{q}_i$ and document vector $\mathbf{d}_j$, and

$$M^*(i,j) = min\left(M_{i-1,j}, M_{i-1,j-1}, M_{i,j-1}\right) \tag{12}$$

The matrix computed following Eq. (11) is a cumulative cost matrix, where the cost at each position (i, j) takes into account the cost at this point and also the cost at the previous steps. Following the restrictions of the DTW algorithm, the alignment path can move in three different directions, as represented in Eq. (12): one step horizontally, one step vertically, or one step horizontally and

vertically at the same time. Since DTW aims at minimizing the cost, Eq. (12) selects the previous step as the one with the smallest cost among these three alternatives.

In this work, the negative log cosine similarity was used as the cost function $cost(\mathbf{q}_i, \mathbf{d}_j)$, since it is a suitable alternative when dealing with phone posteriorgrams [69]:

$$cost(\mathbf{q}_i, \mathbf{d}_j) = -log \frac{\mathbf{q}_i \cdot \mathbf{d}_j}{|\mathbf{q}_i| \cdot |\mathbf{d}_j|} \tag{13}$$

$cost(\mathbf{q}_i, \mathbf{d}_j)$ is normalized in order to turn it into a cost function defined in the interval [0,1] as follows [67]:

$$c(\mathbf{q}_i, \mathbf{d}_j) = \frac{cost(\mathbf{q}_i, \mathbf{d}_j) - cost_{min}(i)}{cost_{max}(i) - cost_{min}(i)} \tag{14}$$

where $cost_{min}(i) = \min_j cost(\mathbf{q}_i, \mathbf{d}_j)$ and $cost_{max}(i) = \max_j cost(\mathbf{q}_i, \mathbf{d}_j)$. Therefore, $c(\mathbf{q}_i, \mathbf{d}_j)$ is a normalized cost function derived from the cosine similarity.

Once the matrix $\mathbf{M}$ is obtained, the best alignment path between a query Q and a document D (i.e. the sequence of steps that leads to the minimum alignment cost between Q and D) can be obtained using the S-DTW algorithm. First, the last step of the best alignment path b^* is selected as the lowest cumulative cost of all the possible ones:

$$b^* = \underset{b \in 1,\ldots,n_D}{\arg\min}\ M_{n_Q,b} \tag{15}$$

Since M is a cumulative matrix cost, each element $M_{n_Q,b}$, $b \in 1,\ldots,n_D$ in the last row of the matrix represents the cost of ending the path at position b. Therefore, the last step of the path with the lowest cost can be found by searching for the value of b that minimizes the cost, as defined in Eq. (15). Then, the first step a^* can be obtained by backtracking the path starting at b^*. This results in an alignment path

$$Path(Q, D) = \{p_1, \ldots, p_k, \ldots, p_K\} \tag{16}$$

where $p_k = (i_k, j_k)$, i.e. the k^{th} step of the path is formed by $\mathbf{q}_{i_k}$ and $\mathbf{d}_{j_k}$.

4.3. Score normalization

The search stage returns, for each query-document pair, the minimum alignment cost M_{n_Q,b^*}, which is the minimum cumulative cost resulting from aligning query Q and document D. This value can be interpreted as a score that indicates how reliably the query was found in the document. Nevertheless, this cost strongly depends on the length of the document and the query, so length normalization is usually applied to this value [65]:

$$score(Q, D) = \frac{M_{n_Q,b^*}}{b^* - a^* + n_Q} \tag{17}$$

This normalization is equivalent to dividing the score by the length of the best alignment path, estimated as the number of matching frames in the document ($b^* - a^*$) plus the number of frames in the query n_Q.

Afterwards, as explained in Section 3, it is necessary to make the scores of different queries comparable among them, since a decision threshold must be applied to decide whether a query was present or not in a document. Hence, in this system, the z-norm defined in Eq. (8) was also applied to the scores.

5. Combination of phone multigram and DTW systems

This section describes two different combinations of the phone multigram and DTW systems described in Sections 3 and 4, respectively. The first strategy, namely two-stage approach, aims at obtaining a system with less computational cost than the pure DTW strategy by reducing the number of query-document pairs to be evaluated by this algorithm. The second strategy, namely fusion approach, aims at improving the individual performance of the phone multigram and DTW systems by combining their output scores.

5.1. Two-stage approach

As mentioned in Section 1, DTW-based approaches for QbESDR are effective but their computational cost is high: given a query and a set of documents, a cost matrix must be computed and the best alignment path must be found

for each query-document pair. A two-stage approach is proposed in this work, which aims at using a computationally efficient strategy to select a reduced set of query-document pairs to be evaluated by the DTW strategy. In this way, the overall computational cost of the search stage is reduced, since the number of DTW evaluations decreases.

In this proposed combination, first search using the phone multigram approach is performed, which results in a score for each query-document pair. Since higher scores represent a higher probability of having found the query in the document, those query-document pairs that have a score above a given threshold are selected as preliminary candidates. Since the score normalization strategy described in Section 3 leads to scores with zero mean and unit variance, scores above 0 (i.e. above the mean) are considered as candidates.

Once the set of candidate query-document pairs are identified by means of the phone multigram approach, they are evaluated using the DTW-based strategy described in Section 4. This two-stage approach leads to a reduction of the computational cost proportional to the number of selected candidates on the first stage.

5.2. Fusion approach

The phone multigram and DTW approaches for QbESDR have the same functionality (i.e. searching for queries within spoken documents), but this is achieved by means of very different approaches. Hence, it is straightforward to combine these different experts in order to enhance their individual performance. A strategy based on score calibration and fusion is proposed for this purpose, which is a common approach in QbESDR [46, 25, 63].

As described in [46], first missing scores must be hypothesized, since it might happen that one of the systems has not succeeded at outputting a score for a query-document pair. This is usually done by assigning the minimum global score to those query-document pairs. Then, calibration and fusion parameters must be estimated in a training set, which can be done through logistic regression as described in [46]. Once the fusion parameters are obtained, it is possible

to combine the scores of different systems by means of a pooled weighted sum:

$$score^f(Q,D) = \beta + \sum_{i=1}^{S} \alpha_i \cdot score^i(Q,D) \tag{18}$$

where S is the number of systems to combine, $score^f(Q,D)$ is the resulting fused score for query Q and document D, $score^i(Q,D)$ is the score output by system i for query Q and document D, and α_i and β are the fusion parameters.

This type of fusion aims at increasing the performance achieved by the combined systems, but this occurs at the expense of increasing the computational cost of the search stage. Nevertheless, since the phone multigram approach is computationally efficient (especially compared to the DTW strategy), only a slight reduction of the efficiency is observed.

6. Experimental framework

The framework used in MediaEval 2014 Query-by-Example Search on Speech (QUESST 2014) evaluation [18] was employed in this work, since it represents a challenging real-life scenario for QbESDR. QUESST 2014 database includes a set of audio documents where the search must be performed, a set of development (dev) queries for system training, and a set of evaluation (eval) queries to assess the performance after training, as summarized in Table 1. The audio documents include speech in six different languages, namely Albanian, Basque, Czech, non-native English, Romanian, and Slovak. These documents were collected from multiple sources such as broadcast news programs, telephone calls into radio live broadcasts, TED talks or Parliament meetings [70]. Hence, the database features read and spontaneous speech as well as broadcast speech and lectures, and there are mismatched acoustic conditions since the data includes clean and noisy speech. The queries, which feature the six aforementioned languages, were recorded using a mobile phone in order to simulate a regular user querying a retrieval system via speech [70]. There are three different matching types in this experimental framework:

- Exact (T1): a hit is produced when an exact match of the lexical representation of the query is found in a document.
- Variant (T2): hits allow slight variations of the lexical representation of the query either at the beginning or at the end of the query. For example, "engineer" should match a document saying "engineering" and *vice versa*.
- Reordering/filler (T3): given a query with multiple words, a hit is produced when the document contains all the words in the query but they might appear in a different order and/or with a small amount of filler content between words. Lexical variations as in T2 queries are also allowed. For example, "Brazilian president" should match a document saying "president of Brazil".

Some statistics about the queries are summarized in Table 2. It must be noted that one query can belong to more than one matching type: for example, the query "engineer" can be of matching type T1 and T2 if there are document hits that are exact matches and also others that imply lexical variations. It should be mentioned that, in this table, the sum of the number of hits of matching types T1, T2 and T3 is not equal to the number of hits of All matching types: following the previous example, a document containing the words "engineer" and "engineering" is a hit for types T1 and T2 of the query "engineer" (i.e. two hits) but it counts as a single hit for All.

Two evaluation metrics defined in the experimental protocol of QUESST 2014 were used in this work to assess search on speech performance and computational cost.

QbESDR performance is evaluated by means of the maximum term weighted value (MTWV) [71], which is derived from the term weighted value (TWV). Let $\mathcal{Q}$ be a set of $|\mathcal{Q}|$ queries that must be searched within a set of documents Ω. The TWV aims at measuring the amount of actual query matches that were not found in Ω (miss detections) and the amount of false query matches that were detected by the system (false alarms). For this purpose, a decision threshold θ is used to decide whether a score represents a match of the query in the document.

Table 1: Summary of the QbESDR experimental framework used in this paper: number of recordings in each set (# recordings); total (Total), minimum (Min) and maximum (Max) duration of the recordings. Audio docs represent the spoken documents were the search must be performed, and dev/eval queries represent the sets of queries for system training and testing, respectively.

		Duration		
Data	# recordings	Total	Min	Max
Audio docs	12492	23 h 5 min	0.63 s	47.17 s
dev queries	560	20 min 22.92 s	0.56 s	6.18 s
eval queries	555	19 min 27.61 s	0.52 s	3.62 s

Table 2: Summary of the query-by-example spoken document retrieval (SDR) experimental framework used in this paper. Matching type denotes the type of matching (All - all matching types, T1 - exact, T2 - variant, T3 - reordering/filler), query set represents the set of queries (dev, eval), # queries stands for the number of queries in each set, and # hits represents the number of retrieved documents for each set.

Matching type	Query set	# queries	# hits
All	dev	560	5471
	eval	555	5213
T1	dev	307	2102
	eval	307	2084
T2	dev	190	2450
	eval	179	2180
T3	dev	155	1026
	eval	156	1068

Formally, TWV is defined as the complement of the measurement of false alarms and miss detections:

$$TWV(\theta) = 1 - \frac{1}{|\mathcal{Q}|} \sum_{\forall Q \in \mathcal{Q}} \{P_{miss}(Q,\theta) \ + \beta \cdot P_{fa}(Q,\theta)\} \tag{19}$$

where $P_{miss}(Q,\theta)$ is the probability of missing hits of Q given θ, $P_{fa}(Q,\theta)$ is the probability of inserting false hits of Q given θ, and the weight factor β is

defined as:

$$\beta = \frac{C_{fa}(1 - P_{target})}{C_{miss}P_{target}} \tag{20}$$

where $C_{miss} > 0$ and $C_{fa} > 0$ are the costs of miss and false alarm errors, respectively, and P_{target} is the prior probability of finding a match of a query in a document (which is assumed to be constant across queries).

The MTWV is defined as the TWV at the optimal decision threshold θ_{opt} (i.e. the decision threshold that leads to the maximum value of TWV given the scores computed by the system):

$$MTWV = TWV(\theta_{opt}) \tag{21}$$

The MTWV was computed using the official evaluation tool of QUESST 2014. It must be noted that the values of C_{fa}, C_{miss} and P_{target} were fixed in the evaluation protocol of QUESST 2014 to 1, 100 and 0.0008, respectively.

The computational cost is measured by means of the searching speed factor [72]:

$$SSF(\mathcal{Q}, \Omega) = \frac{T_{Searching}}{T_{\mathcal{Q}} \cdot T_{\Omega}} \tag{22}$$

where $T_{Searching}$ is the time in seconds required for searching for the queries in $\mathcal{Q}$ within the set of documents Ω, and $T_{\mathcal{Q}}$ and T_{Ω} are the total durations in seconds of the sets of queries $\mathcal{Q}$ and documents Ω, respectively. Given an experiment, its SSF was obtained by averaging the $T_{Searching}$ observed in ten executions of the experiment.

7. Experimental results

This Section describes the experimental results obtained in a series of experiments. First, the training of the phone multigram system is done by carrying out different experiments varying the parameters of the system. Afterwards, two different combinations of this approach with a DTW-based system are assessed.

Since the phone multigram representation relies on ASR transcriptions and the DTW-based system uses phone posteriorgrams for speech representation, a phone decoder has to be used in both systems. In these experiments, the phone

decoders developed by the Brno University of Technology (BUT) [73] were used, as they are frequently employed in search on speech tasks, and their use allows the reproducibility of the experiments and the comparison between different approaches. The default configuration parameters were used which, in the case of phone posteriorgrams, led to speech frames of 25 milliseconds extracted every 10 milliseconds.

It must be mentioned that, besides the phone units, these decoders include several silence and noise units. In the phone posteriorgram experiments, these silence and noise units where combined into a single silence/noise unit, given that the specific type of sound is irrelevant for the task. In addition, the silence/noise units were removed from the queries but kept in the documents. The occurrence of silence/noise units is common at the beginning and the end of the queries, but they are unlikely to appear in central positions, since queries are not long enough for requiring pauses when uttering them. Nevertheless, the documents are longer and silence/noise units help to split the documents into sentences, so keeping the silence/noise units in this case helps to avoid mixing phones from different sentences within the same phone n-gram. In the DTW experiments, where the phone decoder is used to obtain phone posteriorgrams, first the posterior probabilities of the silence and noise units were averaged and, in case the posterior probability of this unit was greater than all those corresponding to phone units, the frame was considered as silence/noise and subsequently removed, as done in [67].

To foster reproducibility, the source code required for executing the experiments described in this section is provided[2].

7.1. Phone multigram training

A series of experiments were run in order to analyze the performance of the phone multigram representation according to the size of the n-grams, the number of transcription hypotheses used for the queries and the search time

[2] `http://irlab.org/files/multigram_dtw_ipm2018.zip`

observed dependent on this latter parameter. Specifically:

- A comparison between phone n-gram systems and the phone multigram approach was performed.
- The influence of the number of query transcriptions hypotheses n_{hyp} on system performance was assessed.
- Lastly, the influence of parameters min_{ngram} and max_{ngram} was evaluated.

The Czech (CZ) phone decoder from BUT was used in these training experiments since it empirically showed better results in the dataset used in this work. This phone decoder includes 42 phone units and 3 silence/noise units, that are further combined into a single silence/noise unit.

First, the influence of using phone n-grams of different size, with $n \in \{1, \ldots, 5\}$, when searching for a single query hypothesis ($n_{hyp} = 1$) was assessed and compared with the phone multigram representation with $min_{ngram} = 1$ and $max_{ngram} = 5$. These results are shown in Figure 3 in terms of their MTWV, and results are displayed for All, T1, T2 and T3 matching types. The figure shows that the best results using a single n-gram representation are achieved with 3-grams. When the n-gram size is greater, results decline probably due to the transcription errors: the longer the n-gram, the least likely to be found in the documents. The results using 1-grams are poor as expected, since the positional information is not taken into account and such terms are very easy to match. Figure 3 also shows that the 3-gram system is clearly outperformed by the proposed phone multigram strategy, which achieves a relative improvement by 24% when considering All matching types. It can also be observed that the phone multigram approach achieves the best results for matching types T1, T2 and T3.

An analysis of the influence of the number of hypotheses of the query transcription n_{hyp} was also performed. Figure 4 depicts the performance achieved when varying n_{hyp}: these results show a dramatic improvement of MTWV when

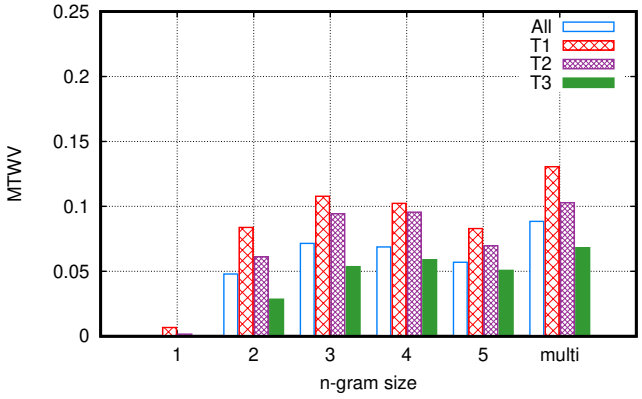

Figure 3: MTWV on the dev experiment when using the CZ phone decoder for All, T1, T2 and T3 matching types using one query hypothesis. Results are shown for phone n-grams with $n \in \{1, \ldots, 5\}$ and for the proposed phone multigram representation with $min_{ngram} = 1$ and $max_{ngram} = 5$.

using several hypotheses compared to using just one. Comparing the results obtained with $n_{hyp} = 150$ to those achieved with a single candidate per query, relative improvements by 77%, 69%, 55% and 57% are observed for All, T1, T2 and T3 matching types, respectively. Nevertheless, when comparing the results achieved with $n_{hyp} = 150$ with those obtained with $n_{hyp} = 400$, the relative improvement when considering All matching types is only 4%. At this point, it is important to analyze the trade-off between performance and search time, since increasing n_{hyp} means linearly increasing the search time, as shown in Figure 5. This figure displays the SSF observed when searching all the dev queries dependent on the value of n_{hyp}, along with the corresponding performance in terms of MTWV. The figure shows that, while the SSF linearly increases with n_{hyp}, the MTWV stops improving to a great extent when using about 150 hypotheses. As mentioned, the MTWV obtained when using 400 hypotheses achieves a relative improvement in performance by 4% with respect to the results obtained with 150 hypotheses, but at the expense of increasing the SSF by 34%. Hence, from now on, n_{hyp} is fixed to 150, since this working point achieves a good trade-off between search time and performance.

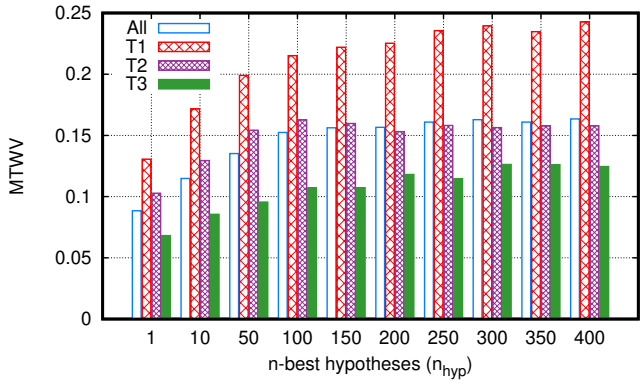

Figure 4: MTWV on the dev queries when using the CZ phone decoder for All, T1, T2 and T3 matching types dependent on the number of n-best query hypotheses n_{hyp}. Results are shown for the proposed phone multigram approach with $min_{ngram} = 1$ and $max_{ngram} = 5$.

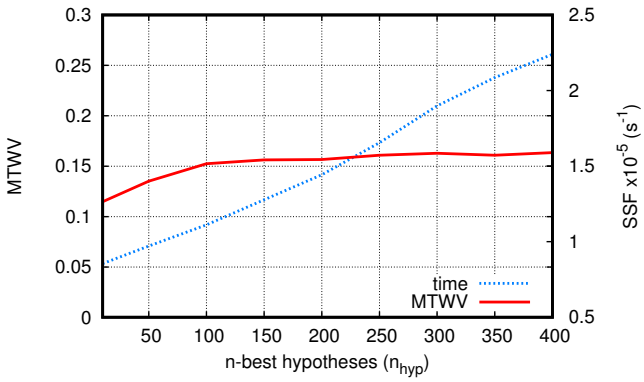

Figure 5: MTWV and SSF (multiplied by 10^{-5} for clarity purposes) obtained when searching all the dev queries dependent on the number of query hypotheses n_{hyp}. These results were computed using the CZ phone decoder, $min_{ngram} = 1$ and $max_{ngram} = 5$.

The previous experiments were run using the phone multigram representation considering $min_{ngram} = 1$ and $max_{ngram} = 5$. Table 3 displays the MTWV achieved when varying min_{ngram} and max_{ngram} from 1 to 10. As shown, the best results are obtained with $min_{ngram} = 1$ and $max_{ngram} = 5$. Longer n-grams do not contribute to improving performance since, as hypothesized above, bigger n-grams are less likely to exactly match the documents. This is easily noticeable when looking at the diagonal of Table 3 (which shows the results for single n-gram systems, since $min_{ngram} = max_{ngram}$): it can be seen that, for n-grams with n bigger than 3, the performance starts to decline. This, combined with the scoring strategy used in this system, explain the negative impact of long n-grams in the performance of the phone multigram approach: the greater the number of matching terms in the query, the greater the score (due to the coordination factor present in Eq. (2)), so adding query terms that are unlikely to be found reduces overall performance. It can also be noted from Table 3 that, while 1-grams were not considered in [30] and [31], adding them to the phone multigram approach yields a relative improvement by 1.5% with respect to considering only 2-grams to 5-grams.

Table 3: MTWV on the dev queries using the CZ phoneme decoder dependent on the values of min_{ngram} and max_{ngram}.

		max_{ngram}									
		1	2	3	4	5	6	7	8	9	10
min_{ngram}	1	0.0018	0.0587	0.1330	0.1506	**0.1562**	0.1557	0.1541	0.1529	0.1521	0.1509
	2	-	0.0749	0.1436	0.1534	0.1538	0.1519	0.1484	0.1474	0.1452	0.1450
	3	-	-	0.1490	0.1529	0.1445	0.1418	0.1376	0.1359	0.1363	0.1365
	4	-	-	-	0.1345	0.1328	0.1322	0.1310	0.1306	0.1308	0.1306
	5	-	-	-	-	0.1297	0.1237	0.1252	0.1263	0.1265	0.1266
	6	-	-	-	-	-	0.0837	0.0883	0.0883	0.0883	0.0883
	7	-	-	-	-	-	-	0.0386	0.0430	0.0430	0.0430
	8	-	-	-	-	-	-	-	0.0179	0.0209	0.0209
	9	-	-	-	-	-	-	-	-	0.0054	0.0086
	10	-	-	-	-	-	-	-	-	-	0.0018

7.2. Testing of phone multigram and combination strategies with DTW

After tuning the system parameters, two experiments were conducted with different objectives:

- Compare the performance of the phone multigram strategy with a state-of-art DTW-based approach.
- Combine both strategies by means of the two-stage and fusion approaches described in Section 5, in order to obtain systems with enhanced performance compared to the phone multigram and DTW-based approaches individually.

Two different phone decoders were used in these experiments, namely the CZ decoder and the Hungarian (HU) decoder from BUT. The purpose of these experiments was to observe whether a similar trend is followed when using different decoders. The HU decoder has 58 phone units and 3 silence/noise units that were managed in the same manner as in the CZ decoder.

Table 4 presents, for both CZ and HU decoders, the results achieved on the eval queries with the DTW-based QbESDR system described in Section 4 (first row), and with the phone multigram representation (second row). The table shows that the DTW system achieves superior results compared to the phone multigram representation except for matching type T3, where the phone multigram strategy achieves a slightly better performance. As mentioned in Section 1, improving the performance for non-exact matches was one of the expected advantages of this method. In addition, comparing the SSF of both systems, summarized in Table 5, it is shown that the time demands of the phone multigram system are smaller than that of the DTW system in several orders of magnitude. Hence, given that one system is efficient and the other is effective, combining both systems to take advantage of their individual strengths seems straightforward.

The two-stage combination strategy aims at obtaining an efficient DTW-based QbESDR system by reducing the number of query-document pairs that

must be evaluated with the DTW approach. Table 4 shows that the results of the two-stage system are not significantly different to those of the DTW approach. These results suggest that the two-stage approach is quite advantageous, since it achieves the same performance as the DTW strategy while readucing its search time by 34% for the CZ decoder, and by 35% when using the HU decoder (this is the effect of reducing the total number of candidates by 66% and 65%, respectively). These results suggest that using the phone multigram approach as a previous step for DTW reduces the computational cost of the resulting system to a great extent while achieving the same performance as the DTW strategy.

The purpose of the fusion approach is to obtain a QbESDR system that increases the individual efficacy of phone multigram and DTW strategies by combining their outputs: given that the search time of the phone multigram strategy is almost negligible compared to that of the DTW system, running both systems would not lead to a significant increase of the search time. In this experiment, Bosaris toolkit [74] was used for that purpose, and calibration and fusion parameters were estimated on the scores of the dev queries and subsequently applied to the scores of the eval queries [75]. The obtained results of this fusion strategy (fourth row in Table 4) show a huge performance improvement when combining both approaches, which suggest that they are strongly complementary.

It must be noted that the same trend is observed when using CZ and HU decoders, which suggests that this approach is not strongly dependent on the phone units used for document and query transcription. Table 5 shows that the SSF values measured for the HU decoder are slightly superior to those observed for the CZ decoder, which is due to the greater number of units in the HU decoder (42 versus 58). It is noticeable that the CZ decoder leads to much better results than the HU decoder. This is probably caused by a simple reason: the database used in these experiments includes Czech speech, so the use of a matching phone decoder boosts the performance on this language. In addition, some of the other languages included in the database (such as Romanian and,

especially, Slovak) are much closer to Czech than to Hungarian, and having a greater phonetic similarity makes the CZ phone decoder more suitable for their representation. As analyzed in [48], some phone units included in the decoders might suit a given language but act as a nuisance for a different one. Apart from the similarity among the languages, the Hungarian decoder has a bigger number of phone units, which increases the confusability among phones and may lead to worse results.

Table 4: MTWV on the eval queries using the CZ and HU phone decoders for four different systems: a DTW-based approach (DTW); the proposed multigram representation (Multigram); a system based on DTW where a selection of match candidates is done using the phone multigram system (two-stage); a combination of the DTW and phone multigram systems by fusion of the retrieved scores (fusion). Fusion results with superindices a, b and c show a statistically significant improvement over DTW, multigram and two-stage systems, respectively. Two-stage results with superindices x and y show a statistically significant improvement over DTW and multigram systems, respectively. Statistical significance was computed based on a t-test ($p < 0.05$).

		MTWV			
Decoder	System	All	T1	T2	T3
CZ	DTW	0.2603	0.4344	0.1905	0.0674
	Multigram	0.1735	0.2529	0.1461	0.1573
	Two-stage	0.2669^{y}	0.4457^{y}	0.1943^{y}	0.0782^{y}
	Fusion	$\mathbf{0.3379}^{a,b,c}$	$\mathbf{0.5041}^{a,b,c}$	$\mathbf{0.2739}^{a,b,c}$	$\mathbf{0.1945}^{a,b,c}$
HU	DTW	0.2260	0.3749	0.1691	0.0541
	Multigram	0.1077	0.1612	0.0817	0.1111
	Two-stage	0.2199^{y}	0.3661^{y}	0.1650^{y}	0.0579^{y}
	Fusion	$\mathbf{0.2949}^{a,b,c}$	$\mathbf{0.4309}^{a,b,c}$	$\mathbf{0.2280}^{a,b,c}$	$\mathbf{0.1653}^{a,b,c}$

8. Review of other approaches

Given that QUESST 2014 is a strict experimental benchmark with well-defined training and test experiments, it is possible to perform comparisons of

Table 5: Searching speed factor (SSF) computed on the eval experiment of DTW, multigram, two-stage and fusion systems.

	SSF	
System	CZ	HU
DTW	$4.00 \cdot 10^{-2}$	$4.19 \cdot 10^{-2}$
Multigram	$1.42 \cdot 10^{-5}$	$1.46 \cdot 10^{-5}$
Two-stage	$1.33 \cdot 10^{-2}$	$1.40 \cdot 10^{-2}$
Fusion	$4.00 \cdot 10^{-2}$	$4.19 \cdot 10^{-2}$

the results displayed in the previous section with others found in the literature. Hence, this section presents a comparison of the approaches presented in this paper with other reported results for the QUESST 2014 dataset. First, the phone multigram representation is compared to other approaches based on phone transcriptions [5, 6]. Then, the two-stage and fusion combinations are compared with similar ones proposed in [49, 6].

An approach based on phone transcriptions for QbESDR, namely symbolic search (SS), is proposed in [5], which consists in looking for subsequences of the phone transcription of a query within the phone lattices of the documents. Equally to the approach proposed in this work, several query transcription hypotheses are considered in this work but, in [5], weighted finite state transducer (WFST)-based search is done on phone lattices, which leads to higher search time and larger indices. Comparing the results presented in [5] with those displayed in Table 4, it can be seen that the phone multigram representation using the CZ decoder succeeds to outperform all the results presented in that work. In addition, the results for matching types T2 and T3 using the CZ decoder outperform those reported in [5] to a great extent, which was, indeed, one of the objectives of this strategy. It must be noted that the CZ and HU decoders used in [5] are exactly the same as those used to obtain the results in Table 4, which suggests that the phone multigram representation leads to a better exploitation of phone transcriptions. With respect to the computational cost of

the algorithms, a comparison between the search times presented in Table 5 and those mentioned in [5] suggests that the phone multigram system is much faster, even though it is not possible to straightforwardly compare those times since the machines used to compute them are different. Nevertheless, this difference in search time might be explained by the use of more complex indices in [5] and by the difference on the number of hypotheses for the queries: 150 hypotheses were used to obtain the results in Table 4 versus 2^{n_p} in [5], where n_p is the average number of phones of the first 1000 n-best hypotheses (the median of this value is around 10 phones, which leads to 1024 hypotheses).

More results based on the SS system proposed in [5] were presented in [6], where a different phone decoder was used to obtain the transcriptions. The phone multigram representation outperforms the results displayed in that work even though the phone decoder used in [6] is more suitable for this task than the CZ decoder used in the experiments presented in this paper (this can be hypothesized by comparing the results in [5] and [6]). In addition, 2000 query hypotheses are considered in [6], which probably led to a higher search time than that reported in [5]. This comparison suggests that the phone multigram system would experiment a performance boost when using more suitable phone decoders.

A combination of SS and DTW systems is also presented in [6]. The obtained results are almost the same as those achieved by the fusion strategy proposed in this paper but the computational cost is greater: first SS must be peformed and afterwards, each query subsequence that produced a match in the SS stage is considered as a different query for the DTW stage, which implies multiplying the search time of the DTW stage by the number of selected subsequences (according to [6], all the subsequences of six phones found in the three best query hypotheses are used, which implies, minimum, multiplying the search time by three).

A strategy which is not based on phone decoding but has some similarities with that presented in this work is the BoAW approach proposed in [57], which was applied to the QUESST 2014 evaluation framework in [49]. This approach

first performs phonetic segmentation of the queries and documents by means of the spectral transition measure [76], and then each segment is assigned to a class, i.e. a phone unit. This leads to a phone-like representation but, in this case, the phone units are not obtained from a phone decoder but from automatic clustering of the discovered units. Then, cosine similarity and tf-idf are used to select query-document pairs, which are further evaluated using a DTW system. Results using only the BoAW model are not displayed in [49], so it is not possible to compare this strategy with the phone multigram representation. Nevertheless, results when combining BoAW and DTW as in the two-stage strategy described in Section 5 are presented in that work. At first sight, the performance of BoAW+DTW is superior to that of the two-stage approach, but some issues must be considered. First, the DTW system used in [49] achieved better results than the one described in Section 4 thanks to some refinements of the DTW algorithm presented in that work, so the impact of using a candidate selection strategy (and not the absolute results) must be evaluated to enable a fair comparison. Second, the number of selected candidates ranges from 50% to 100% of the total in [49] depending on the decision threshold applied to the scores output by the BoAW system; in the two-stage system, a fixed threshold was used, which reduces the number of system parameters to be tuned while leading to a lower number of candidates (34% of the total candidates were kept in the results displayed in Table 4 for the CZ decoder). With that in mind, it can be observed that the proposed two-stage approach proposed in this work leads to the same results as the DTW strategy described in Section 4, while using BoAW with a pruning threshold by 50% dramatically degrades system performance compared to the DTW results reported in [49].

9. Conclusions and future work

This paper presented a novel representation for QbESDR based on a combination of different-sized n-grams, namely phone multigrams, obtained from automatic phone transcriptions of the queries and documents. This represen-

tation, which uses several query transcription hypotheses to ease the influence of transcription errors, is used for search within inverted indices, which dramatically reduces the search time for real-world QbESDR applications. The performance of this system was assessed in the framework of MediaEval 2014 Query-by-Example Search on Speech (QUESST 2014) evaluation, and the experimental results show that the proposed approach outperforms other similar approaches found in the literature while dramatically reducing the search time. In addition, a two-stage strategy was proposed in which the proposed phone multigram strategy is used to search for candidate matches of the queries in the documents, and these candidates are further evaluated using a DTW-based approach. This combination led to the same performance as the DTW approach while reducing the search time to a great extent. Also, a fusion strategy was proposed in which the scores of the phone multigram and DTW-based systems were combined, and this combination proved to boost the performance with respect to both methods individually while exhibiting a negligible increase of the search time.

The proposed phone multigram system used 1-best transcriptions of the documents in order to reduce the size of the indices and the search time; in future work, other alternatives for document representation will be explored, aiming at keeping the size of the index as small as possible while improving the performance of the method proposed in this work. Also, the use of the confidence scores output by the phone decoder will be investigated in order to assign higher scores to those matches of query-document pairs whose transcriptions were considered as more reliable by the decoder.

The performance analyses presented in this paper showed the impact of the suitability of the phone decoder on the results: as expected in cross-lingual approaches, the more similar the language of the decoder and the documents, the better the performance. Different approaches will be explored in the future to try to optimize the performance of the decoder independently of the language of the documents and queries. Specifically, phone selection and clustering will be investigated.

This paper explored the use of the multigram approach for spoken document retrieval using spoken queries. Nevertheless, in future work, the use of this strategy using written queries will be explored, especially for searching out-of-vocabulary words.

Acknowledgements

This work has received financial support from i) "Ministerio de Economía y Competitividad" of the Government of Spain and the European Regional Development Fund (ERDF) under the research project TIN2015-64282-R, ii) Xunta de Galicia (project GPC ED431B 2016/035), and iii) Xunta de Galicia - "Consellería de Cultura, Educación e Ordenación Universitaria" and the ERDF through the 2016-2019 accreditation ED431G/01 ("Centro singular de investigación de Galicia").

References

[1] K. Sparck Jones, G. Jones, J. Foote, S. Young, Experiments in spoken document retrieval, Information Processing & Management 32 (4) (1996) 399–417.

[2] S. Nakagawa, K. Iwami, Y. Fujii, K. Yamamoto, A robust/fast spoken term detection method based on a syllable n-gram index with a distance metric, Speech Communication 55 (3) (2013) 470–485.

[3] N. Sakamoto, K. Yamamoto, S. Nakagawa, Spoken term detection based on a syllable n-gram index at the NTCIR-11 Spoken Query&Doc task, in: Proceedings of the 11th NTCIR Conference, 2014, pp. 419–424.

[4] M. Martinez, P. Lopez-Otero, R. Varela, A. Cardenal-Lopez, L. Docio-Fernandez, C. Garcia-Mateo, GTM-UVigo systems for Albayzin 2014 search on speech evaluation, in: Iberspeech 2014: VIII Jornadas en Tecnología del Habla and IV SLTech Workshop, 2014.

[5] H. Xu, P. Yang, X. Xiao, L. Xie, C.-C. Leung, H. Chen, J. Yu, H. Lv, L. Wang, S. Leow, B. Ma, E. Chng, H. Li, Language independent query-by-example spoken term detection using n-best phone sequences and partial matching, in: Proceedings of the 37th International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2015, pp. 5191–5195.

[6] H. Xu, J. Hou, X. Xiao, V. Pham, C.-C. Leung, L. Wang, V. Do, H. Lv, L. Xie, B. Ma, E. Chng, H. Li, Approximate search of audio queries by using DTW with phone time boundary and data augmentation, in: Proceedings of the 37th International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2016, pp. 6030–6034.

[7] C. van Heerden, D. Karakos, K. Narasimhan, M. Davel, R. Schwartz, Constructing sub-word units for spoken term detection, in: Proceedings of the 42nd International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2017, pp. 5780–5784.

[8] H. Sakoe, S. Chiba, Dynamic programming algorithm optimization for spoken word recognition, IEEE Transactions on Acoustics, Speech and Signal Processing 23 (1) (1978) 43–49.

[9] M. Müller, Information Retrieval for Music and Motion, Springer-Verlag, 2007.

[10] G. Mantena, S. Achanta, K. Prahallad, Query-by-example spoken term detection using frequency domain linear prediction and non-segmental dynamic time warping, IEEE/ACM Transactions on Audio, Speech and Language Processing 22 (5) (2014) 944–953.

[11] X. Anguera, M. Ferrarons, Memory efficient subsequence DTW for query-by-example spoken term detection, in: Proceedings of IEEE International Conference on Multimedia and Expo (ICME), 2013, pp. 1–6.

[12] X. Anguera, Information retrieval-based dynamic time warping, in: INTERSPEECH, 2013, pp. 1–5.

[13] M. Versteegh, R. Thiolliere, T. Schatz, X. Cao, X. Anguera, A. Jansen, E. Dupoux, The zero resource speech challenge 2015, in: INTERSPEECH, 2015, pp. 3169–3173.

[14] E. Dunbar, X. Cao, J. Benjumea, J. Karadayi, M. Bernard, L. Besacier, X. Anguera, E. Dupoux, The zero resource speech challenge 2017, in: IEEE Workshop on Automatic Speech Recognition & Understanding, ASRU, 2017, pp. 323–330.

[15] F. Metze, N. Rajput, X. Anguera, M. Davel, G. Gravier, C. V. Heerden, G. Mantena, A. Muscariello, K. Pradhallad, I. Szöke, J. Tejedor, The spoken web search task at MediaEval 2011, in: Proceedings of the 37th International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2012, pp. 5165–5168.

[16] F. Metze, E. Barnard, M. Davel, C. V. Heerden, X. Anguera, G. Gravier, N. Rajput, The spoken web search task, in: Proceedings of the MediaEval 2012 Workshop, 2012.

[17] X. Anguera, F. Metze, A. Buzo, I. Szöke, L. Rodriguez-Fuentes, The spoken web search task, in: Proceedings of the MediaEval 2013 Workshop, 2013.

[18] X. Anguera, L. Rodriguez-Fuentes, I. Szöke, A. Buzo, F. Metze, Query by example search on speech at Mediaeval 2014, in: Proceedings of the MediaEval 2014 Workshop, 2014.

[19] I. Szöke, L. Rodriguez-Fuentes, A. Buzo, X. Anguera, F. Metze, J. Proença, M. Lojka, X. Xiong, Query by example search on speech at MediaEval 2015, in: Proceedings of the MediaEval 2015 Workshop, 2015.

[20] T. Akiba, H. Nishizaki, H. Nanjo, G. Jones, Overview of the NTCIR-11 SpokenQuery&Doc task, in: Proceedings of the 11th NTCIR Conference, 2014, pp. 350–364.

[21] T. Akiba, H. Nishizaki, H. Nanjo, G. Jones, Overview of the NTCIR-12 SpokenQuery&Doc-2 task, in: Proceedings of the 12th NTCIR Conference on Evaluation of Information Access Technologies, 2016, pp. 167–179.

[22] J. Tejedor, D. Toledano, X. Anguera, A. Varona, L. Hurtado, A. Miguel, J. Colás, Query-by-example spoken term detection ALBAYZIN 2012 evaluation: overview, systems, results, and discussion, EURASIP Journal on Audio, Speech, and Music Processing 2013 (23).

[23] J. Tejedor, D. Toledano, P. Lopez-Otero, L. Docio-Fernandez, C. Garcia-Mateo, Comparison of ALBAYZIN query-by-example spoken term detection 2012 and 2014 evaluations, EURASIP Journal on Audio, Speech, and Music Processing 2016 (1).

[24] J. Tejedor, D. Toledano, The ALBAYZIN 2016 search on speech evaluation plan, last accessed 9 January 2018 (2016).
URL `https://iberspeech2016.inesc-id.pt/wp-content/uploads/2016/06/EvaluationPlanSearchonSpeech.pdf`

[25] I. Szöke, L. Burget, F. Grézl, L. Ondel, BUT SWS 2013 - massive parallel approach, in: Proceedings of the MediaEval 2013 Workshop, 2013.

[26] P. Yang, H. Xu, X. Xiao, L. Xie, C.-C. Leung, H. Chen, J. Yu, H. Lv, L. Wang, S. Leow, B. Ma, E. Chng, H. Li, The NNI query-by-example system for MediaEval 2014, in: Proceedings of the MediaEval 2014 Workshop, 2014.

[27] J. Hou, V. Pham, C.-C. Leung, L. Wang, H. Xu, H. Lv, L. Xie, Z. Fu, C. Ni, X. Xiao, H. Chen, S. Zhang, S. Sun, Y. Yuan, P. Li, T. Nwe, S. Sivadas, B. Ma, E. Chng, H. Li, The NNI query-by-example system for MediaEval 2015, in: Proceedings of the MediaEval 2015 Workshop, 2015.

[28] J. Proença, L. Castela, F. Perdigão, The SPL-IT-UC query by example search on speech system for MediaEval 2015, in: Proceedings of the MediaEval 2015 Workshop, 2015.

[29] P. Lopez-Otero, L. Docio-Fernandez, C. Garcia-Mateo, GTM-UVigo systems for the query-by-example search on speech task at MediaEval 2015, in: Proceedings of the MediaEval 2015 Workshop, 2015.

[30] S. Harding, W. Croft, C. Weir, Probabilistic retrieval of OCR degraded text using n-grams., in: Research and Advanced Technology for Digital Libraries (ECDL 1997), Vol. 1324 of Lecture Notes in Computer Science, Springer, 1997, pp. 345–359.

[31] J. Parapar, A. Freire, A. Barreiro, Revisiting n-gram based models for retrieval in degraded large collections, in: Proceedings of the 31st European Conference on Information Retrieval Research: Advances in Information Retrieval, Vol. 5478 of Lecture Notes in Computer Science, Springer International Publishing, 2009, pp. 680–684.

[32] D. Can, M. Saraclar, Lattice indexing for spoken term detection, IEEE Transactions on Audio, Speech & Language Processing 19 (8) (2011) 2338–2347.

[33] L. Mangu, H. Soltau, H.-K. Kuo, B. Kingsbury, G. Saon, Exploiting diversity for spoken term detection, in: Proceedings of the 37th International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2013, pp. 8282–8286.

[34] J. Chiu, Y. Wang, J. Trmal, D. Povey, G. Chen, A. Rudnicky, Combination of FST and CN search in spoken term detection, in: Interspeech, 2014, pp. 2784–2788.

[35] A. Buzo, H. Cucu, M. Safta, C. Burileanu, Multilingual query by example spoken term detection for under-resourced languages, in: 7th Conference on Speech Technology and Human-Computer Dialogue (SpeD), 2013.

[36] J. Vilares, M. Alonso, Y. Doval, M. Vilares, Studying the effect and treatment of misspelled queries in cross-language information retrieval, Information Processing & Management 52 (4) (2016) 646–657.

[37] K. Abainia, S. Ouamour, H. Sayoud, Effective language identification of forum texts based on statistical approaches, Information Processing & Management 52 (4) (2016) 491–512.

[38] L. Lopes Figueiredo, G. Tavares de Assis, A. Ferreira, DERIN: a data extraction mehtod based on rendering information and n-gram, Information Processing & Management 53 (2017) 1120–1138.

[39] M. Fatima, K. Hasan, S. Anwar, R. Nawab, Multilingual author profiling on Facebook, Information Processing & Management 53 (2017) 886–904.

[40] V. K., D. Gupta, Unasking text plagiarism using syntactic-semantic based natural language processing techniques: comparisons, analysis and challenges, Information Processing & Management 54 (2018) 408–432.

[41] K. Knill, M. Gales, S. Rath, P. Woodland, C. Zhang, S.-X. Zhang, Investigation of multilingual deep neural networks for spoken term detection, in: IEEE Workshop on Automatic Speech Recognition & Understanding, ASRU, 2013, pp. 138–143.

[42] J. Cui, B. Kingsbury, B. Ramabhadran, A. Sethy, K. Audhkhasi, X. Cui, E. Kislal, L. Mangu, M. Nussbaum-Thom, M. Picheny, Z. Tüske, P. Golik, R. Schlüter, H. Ney, M. Gales, K. Knill, A. Ragni, H. Wang, P. Woodland, Multilingual representations for low resource speech recognition and keyword search, in: IEEE Workshop on Automatic Speech Recognition & Understanding, ASRU, 2015, pp. 259–266.

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

[44] G. Mantena, K. Prahallad, Use of articulatory bottle-neck features for query-by-example spoken term detection in low resource scenarios, in: Proceedings of the 37th International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2014, pp. 7128–7132.

[45] M. Madhavi, H. Patil, VTLN-warped Gaussian posteriogram for QbE-STD, in: Proceedings of 23rd European Signal Processing Conference (EUSIPCO), 2017, pp. 563–567.

[46] A. Abad, L. Rodriguez-Fuentes, M. Penagarikano, A. Varona, G. Bordel, On the calibration and fusion of heterogeneous spoken term detection systems, in: Proceedings of Interspeech, 2013, pp. 20–24.

[47] L. Rodriguez-Fuentes, A. Varona, M. Penagarikano, G. Bordel, M. Diez, High-performance query-by-example spoken term detection on the SWS 2013 evaluation, in: Proceedings of the 37th International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2014, pp. 7869–7873.

[48] P. Lopez-Otero, L. Docio-Fernandez, C. Garcia-Mateo, Phonetic unit selection for cross-lingual query-by-example spoken term detection, in: Proceedings of IEEE Automatic Speech Recognition and Understanding Workshop, 2015, pp. 223–229.

[49] M. Madhavi, H. Patil, Partial matching and search space reduction for QbE-STD, Computer Speech & Language 45 (2017) 58–82.

[50] C. Joder, F. Weninger, M. Wölmer, B. Schuller, The TUM cumulative DTW approach for the Mediaeval 2012 spoken web search task, in: Proceedings of the MediaEval 2012 Workshop, 2012.

[51] M. Calvo, M. Giménez, L. Hurtado, E. Sanchis, J. Gomez, ELiRF at MediaEval 2014: query by example search on speech task (QUESST), in: Proceedings of the MediaEval 2014 Workshop, 2014.

[52] M. Carlin, S. Thomas, A. Jansen, H. Hermansky, Rapid evaluation of speech representations for spoken term discovery, in: Proceedings of Interspeech, 2011, pp. 821–824.

[53] A. Jansen, B. Van Durme, P. Clark, The JHU-HLTCOE spoken web search system for MediaEval 2012, in: Proceedings of the MediaEval 2012 Workshop, 2012.

[54] P. Lopez-Otero, L. Docio-Fernandez, C. Garcia-Mateo, Finding relevant features for zero-resource query-by-example search on speech, Speech Communication 84 (Supplement C) (2016) 24–35.

[55] J. Proença, A. Veiga, F. Perdigão, Query by example search with segmented dynamic time warping for non-exact spoken queries, in: Proceedings of 23rd European Signal Processing Conference (EUSIPCO), 2015, pp. 1691–1695.

[56] J. Proença, F. Perdigão, Segmented dynamic time warping for spoken query-by-example search, in: Proceedings of Interspeech, 2016, pp. 750–754.

[57] B. George, B. Yegnanarayana, Unsupervised query-by-example spoken term detection using segment-based bag of acoustic words, in: Proceedings of the 37th International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2014, pp. 7183–7187.

[58] A. Stolcke, M. Akbacak, L. Ferrer, S. Kajarekar, C. Richey, N. Scheffer, E. Shriberg, Improving language recognition with multilingual phone recognition and speaker adaptation transforms, in: Odyssey 2010: The Speaker and Language Recognition Workshop, 2010.

[59] T. Chia, H. Li, H. Ng, A statistical language modeling approach to lattice-based spoken document retrieval, in: Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, 2007, pp. 810–818.

[60] I. H. Witten, A. Moffat, T. C. Bell, Managing Gigabytes (2nd Ed.): Compressing and Indexing Documents and Images, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1999.

[61] G. Salton, A. Wong, C. Yang, A vector space model for automatic indexing, Communications of the ACM 18 (11) (1975) 613–620.

[62] C. Manning, P. Raghavan, H. Schütze, Introduction to Information Retrieval, Cambridge University Press, 2008.

[63] I. Szöke, L. Burget, F. Grézl, J. Černocký, L. Ondel, Calibration and fusion of query-by-example systems - BUT SWS 2013, in: Proceedings of the 37th International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2014, pp. 7899–7903.

[64] L. Rodriguez-Fuentes, A. Varona, M. Penagarikano, G. Bordel, M. Diez, GTTS systems for the SWS task at MediaEval 2013, in: Proceedings of the MediaEval 2013 Workshop, 2013.

[65] A. Abad, R. Astudillo, I. Trancoso, The L2F spoken web search system for Mediaeval 2013, in: Proceedings of the MediaEval 2013 Workshop, 2013.

[66] I. Szöke, M. Skácel, L. Burget, BUT QUESST2014 system description, in: Proceedings of the MediaEval 2014 Workshop, 2014.

[67] L. Rodriguez-Fuentes, A. Varona, M. Penagarikano, GTTS-EHU systems for QUESST at MediaEval 2014, in: Proceedings of the MediaEval 2014 Workshop, 2014.

[68] A. Varona, M. Penagarikano, L. Rodriguez-Fuentes, G. Bordel, On the use of lattices of time-synchronous cross-decoder phone co-occurrences in a SVM-phonotactic language recognition system., in: INTERSPEECH, 2011, pp. 2901–2904.

[69] B. Gündoğdu, M. Saraçlar, Distance metric learning for posteriorgram based keyword search, in: Proceedings of the 42nd International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2017, pp. 5660–5664.

[70] X. Anguera, L. Rodriguez-Fuentes, A. Buzo, F. Metze, I. Szöke, M. Penagarikano, QUESST2014: evaluating query-by-example speech search in a zero-resource setting with real-life queries, in: Proceedings of the 37th International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2015, pp. 5833–5837.

[71] J. Fiscus, J. Ajot, J. Garofolo, G. Doddington, Results of the 2006 spoken term detection evaluation, in: Proceedings of the ACM SIGIR Workshop "Searching Spontaneous Conversational Speech", 2007, pp. 51–56.

[72] L. Rodriguez-Fuentes, M. Penagarikano, MediaEval 2013 spoken web search task: system performance measures, Tech. rep., Software Technologies Working Group, University of the Basque Country (May 2013).
URL http://gtts.ehu.es/gtts/NT/fulltext/rodriguezmediaeval13.pdf

[73] P. Schwarz, Phoneme recognition based on long temporal context, Ph.D. thesis, Brno University of Technology (2009).

[74] N. Brümmer, E. de Villiers, The BOSARIS toolkit user guide: Theory, algorithms and code for binary classifier score processing, Tech. rep., AGNITIO Research (2011).
URL https://sites.google.com/site/nikobrummer

[75] N. Brümmer, D. van Leeuwen, On calibration of language recognition scores, in: IEEE Odyssey 2006: The Speaker and Language Recognition Workshop, 2006, pp. 1–8.

[76] S. Furui, On the role of spectral transition for speech perception, Journal of the Acoustic Society of America 80 (4) (1986) 1016–1025.