Multimedia Tools and Applications (2020) 79:7927–7949
https://doi.org/10.1007/s11042-019-08522-z

Statistical language models for query-by-example spoken document retrieval

Paula Lopez-Otero[1] · Javier Parapar[1] · Alvaro Barreiro[1]

Received: 7 March 2019 / Revised: 19 November 2019 / Accepted: 26 November 2019 /
Published online: 3 January 2020

Abstract

Query-by-example spoken document retrieval (QbESDR) consists in, given a collection of documents, computing how likely a spoken query is present in each document. This is usually done by means of pattern matching techniques based on dynamic time warping (DTW), which leads to acceptable results but is inefficient in terms of query processing time. In this paper, the use of probabilistic retrieval models for information retrieval is applied to the QbESDR scenario. First, each document is represented by means of a language model, as commonly done in information retrieval, obtained by estimating the probability of the different n-grams extracted from automatic phone transcriptions of the documents. Then, the score of a query given a document can be computed following the query likelihood retrieval model. Besides the adaptation of this model to QbESDR, this paper presents two techniques that aim at enhancing the performance of this method. One of them consists in improving the language models of the documents by using several phone transcription hypotheses for each document. The other approach aims at re-ranking the retrieved documents by incorporating positional information to the system, which is achieved by string alignment of the query and document phone transcriptions. Experiments were performed on two large and heterogeneous datasets specifically designed for search on speech tasks, namely MediaEval 2013 Spoken Web Search (SWS 2013) and MediaEval 2014 Query-by-Example Search on Speech (QUESST 2014). The experimental results prove the validity of the proposed strategies for QbESDR. In addition, the performance when dealing with queries with word reorderings is superior to that exhibited by a DTW-based strategy, and the query processing time is smaller by several orders of magnitude.

✉ Paula Lopez-Otero
paula.lopez.otero@udc.gal

Javier Parapar
javier.parapar@udc.gal

Alvaro Barreiro
alvaro.barreiro@udc.gal

[1] Facultad de Informática, Campus de Elviña S/N, Universidade da Coruña - CITIC, A Coruña, 15071, Spain

Keywords Query-by-example spoken document retrieval · Phone decoding · Phone n-grams · Language models · Minimum edit distance

1 Introduction

The search and retrieval of spoken documents has become an issue of paramount importance due to the proliferation of audiovisual contents that are part of our daily life. This task has created the need for efficient tools to find queries in large sets of documents. Since the use of smartphones has encouraged speech-driven applications, the search of spoken queries has regained the attention of the research community, also boosted by the organization of competitive evaluations related to query-by-example spoken document retrieval (QbESDR), where the aim is to retrieve documents where the query appears [3, 4, 10, 57]; and query-by-example spoken term detection (QbESTD), where the exact position of the query within the document is also required [8, 37, 38, 58–60].

QbESDR is commonly performed following approaches based on either automatic speech recognition (ASR) or pattern matching techniques. The former strategies are inherited from the spoken term detection (STD) task, where queries are formulated in written format and only the documents must be transcribed [13, 15, 32, 43]. However, in QbESDR, queries must be transcribed as well, which adds a new source of noise to the task since both document and query transcriptions might have errors [60]. The performance of ASR-based techniques for QbESDR is reasonable in monolingual scenarios with moderate word error rates [27, 36], and they rely on the availability of language resources to train an ASR system. When these resources are not available for the language of interest, or in multilingual scenarios, it is common to use cross-lingual approaches to obtain subword or word transcriptions using ASR systems trained for a different language [40, 42, 50, 64, 65].

QbESDR techniques based on pattern matching usually rely on the dynamic time warping (DTW) algorithm [51] or any of its variants [6, 7, 34, 39] for finding alignments of the query within the documents. These strategies have exhibited acceptable results in multilingual and language-independent scenarios and they require, in general, fewer resources for training the systems. In these techniques, the queries and documents are represented by means of frame-level feature vectors. Common features are Gaussian posteriorgrams [68], where each feature vector represents the posterior probabilities of a speech frame given a Gaussian mixture model (GMM) [5, 30, 31, 34, 35]; and phone posteriorgrams, where the posterior probability of each phone class in a phone decoder is computed for each frame [2, 24, 25, 29, 49]. Zero-resource representations are also found in the literature, and they consist in extracting features from the waveforms such as Mel frequency cepstral coefficients [12, 21, 36], short-time frequency domain linear prediction features [20], or large sets of features followed by feature selection [26]. The main disadvantage of the QbESDR strategies based on pattern matching is that they are usually inefficient in terms of computational cost [7], which limits their use in practical applications.

Since the performance of current QbESDR approaches still needs improvement, researchers have focused on massive fusions of many different systems in order to boost their individual performance [19, 24, 45, 55, 66] at the cost of increasing the query processing time to a great extent. A strategy for QbESDR based on information retrieval models was presented in [23], which focused on obtaining fast and accurate search systems for real applications. This strategy relies in the phone multigram approach for document and query representation: first, the spoken utterance is transcribed using a phone decoder, and then the sequence of phones is tokenized into n-grams of different sizes, namely phone multigrams.

This representation was used to perform QbESDR from an information retrieval perspective: the documents are stored in an inverted index, and then the queries are searched and scored using the vector space model (VSM) [52] for information retrieval. This system was used for candidate selection, and re-scoring of the selected query-document pairs was done following a DTW-based approach. The experimental results were promising since the performance of this strategy was not significantly different to that of the DTW-based approach while reducing the query processing time by several orders of magnitude. In addition, the performance of the VSM-based system improved to a great extent when dealing with queries with lexical variations and word reorderings.

A new approach for QbESDR is presented in this work, which aims at improving two main aspects of the strategy proposed in [23]: (1) the VSM relies in geometric properties of the document and query representations, and its mathematical derivation includes many heuristics; (2) bag-of-words strategies do not take into account positional information, so there is no guarantee that the n-grams of the query appear in a document in the same order, and finding out the exact position of the query within the document is not trivial.

In this paper, the use of a probabilistic information retrieval model [44, 46, 54] for QbESDR is proposed. This approach gained a great popularity in the information retrieval community since it is more principled than the VSM [33]. Specifically, in this work, language models (LMs) [44] are used to represent documents by means of the probability distribution of their different terms which, in this case, are phone multigrams. Since automatic phone transcriptions usually have errors, a technique to obtain improved LMs for document representation is proposed, which consists in using different transcription hypotheses, instead of the 1-best transcription, to obtain better probability estimates of the different terms in the document. In addition, a re-ranking of the documents is proposed for introducing positional information in this strategy. This re-ranking is done according to the minimum edit distance (MED) between the phone transcription of the query and the document: the optimal query-document alignment is found and a score is computed according to the MED. This score is used to penalize the one obtained from the retrieval model: the penalty is big if the alignment between query and document led to many insertions, deletions and substitutions; on the contrary, the penalty is small if the alignment was successful. This strategy also allows the detection of the exact position of the query within the document, so it makes this method suitable for QbESTD. The validity of the proposed techniques is evaluated for QbESDR and QbESTD and compared with a DTW-based approach in terms of performance and search time.

The rest of this paper is organized as follows: Section 2 presents the QbESDR strategy based on probabilistic retrieval models; Section 3 describes the proposed approaches for improving the LM-based QbESDR strategy; the DTW approach used for comparison is depicted in Section 4; Section 5 summarizes the experimental frameworks; experimental results and a discussion are presented in Section 6; lastly, conclusions and future work are summarized in Section 7.

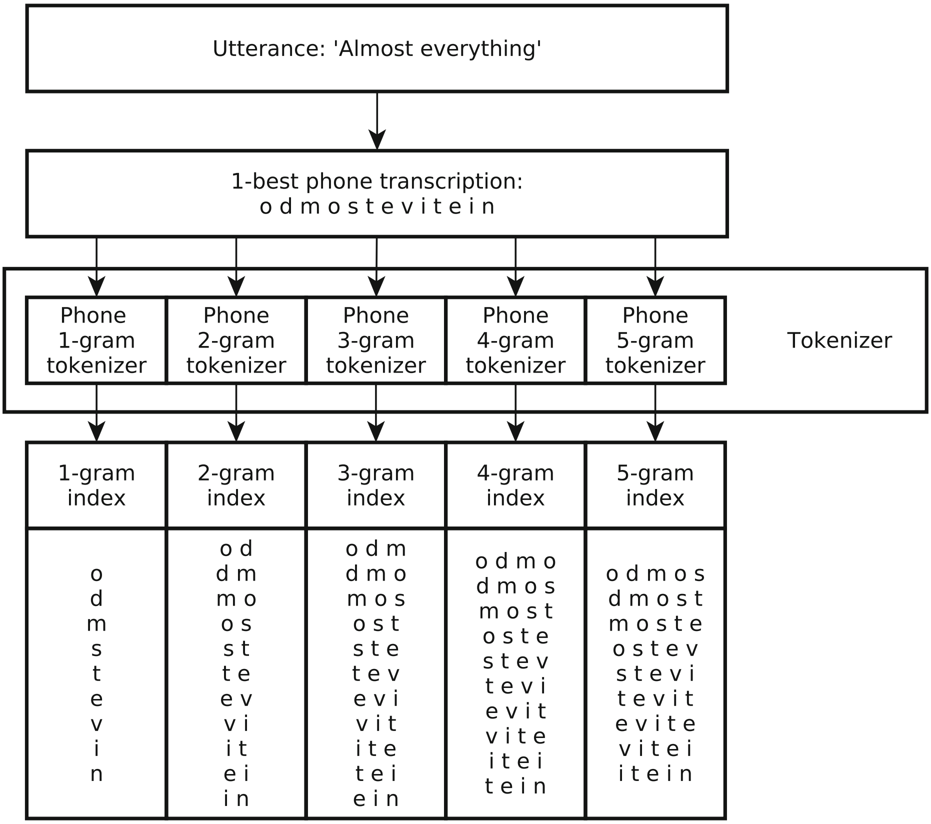
2 Probabilistic information retrieval models for QbESDR

In this work, a probabilistic model for information retrieval is adapted to the QbESDR task. Specifically, the proposed strategy consists in representing documents by means of LMs, which model the probability distribution of the different terms in the document [44]. For this purpose, first a strategy to represent the documents and queries must be chosen, and then indexing and search can be performed. The rest of this section explains these two stages in detail.

2.1 Speech representation

First, a textual representation of the documents and queries must be obtained, and this can be done by means of phone decoding of the audio signals. This procedure converts a speech utterance into a sequence of terms that represent phones, which belong to a set $U = \{u_1, \ldots, u_{n_U}\}$ with n_U phone units. The number of phone units is equal to the number of units in the phone decoding models, as explained in Section 6. Hence, given a speech utterance to transcribe, the phone decoder computes 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 [14]), and different transcription hypotheses, namely n-best transcriptions, can be obtained from this lattice. The 1-best transcription is obtained by finding the most likely sequence of phones according to the probabilities present in the lattice; the 2-best transcription represent the second most likely sequence, and so on.

Once the phone transcription of a speech utterance is extracted, its phone multigram representation can be obtained [23]. This strategy consists in combining different tokenizers for document and query representation: given the phone transcription of a spoken utterance, it is tokenized into n-grams of different sizes, with $n \in \{\min_{ngram}, \ldots, \max_{ngram}\}$, as depicted in the example presented in Fig. 1.



1-gram index	2-gram index	3-gram index	4-gram index	5-gram index
o d m s t e v i n	o d d m m o o s s t t e e v v i i t e i i n	o d m d m o m o s o s t s t e t e v e v i v i t i t e t e i e i n	o d m o d m o s m o s t o s t e s t e v t e v i e v i t v i t e i t e i t e i n	o d m o s d m o s t m o s t e o s t e v s t e v i t e v i t e v i t e v i t e i i t e i n

Fig. 1 Example of the phone multigram approach: the 1-best phone transcription of a speech utterance is obtained and then it is tokenized. In this example, the tokenizer is composed of five phone n-gram tokenizers with $n = 1, \ldots, 5$

2.2 Indexing and search

In text information retrieval, it is common to use inverted indices to store the information related to the documents, since this data structure allows fast and efficient search while achieving an optimal use of storage space [63]. Hence, 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 $D_i = \{t_1, \ldots, t_{n_{D_i}}\}$ obtained from the 1-best transcription of D_i. The inverted index stores, for each present term, a list of all the documents that contain that term. In this work, the terms are phone multigrams as explained above, and they are stored in different indices according to their size, i.e. there is an index for unigrams, another for bigrams, and so forth.

In QbESDR, a score must be assigned to each query-document pair in order to indicate how likely the query matches each document, and this score can be used to decide whether the query is present in the document or not. In LM-based information retrieval systems, scoring is done following the query likelihood retrieval model [33]. Let $Q = \{q_1, \ldots, q_{n_Q}\}$ be a query composed of n_Q terms, which are n-grams of different size (i.e. 1-grams, 2-grams...). For index n (i.e. for n-gram size n), the score of Q given document D is computed as the probability that Q was generated by the language model that represents D:

$$score_{LM}(Q, D, n) = P(Q|D, n) = \prod_{i=1}^{n_Q} P(q_i|D, n) \tag{1}$$

$P(q_i|D, n)$ is the probability that term q_i was generated by the LM of D considering n-grams of size n, which can be computed with the maximum likelihood estimator:

$$P_{ML}(q_i|D, n) = \frac{f_{q_i,D}}{|D|} \tag{2}$$

where $f_{q_i,D}$ is the number of times term q_i appears in document D, and $|D|$ is the total number of tokens in D. The issue regarding this formulation is that, if any of the query terms has $f_{q_i,D} = 0$, then $P(Q|D, n)$ will be zero as well. For this reason, the use of smoothing methods is very common in this framework [67]: given the whole collection of indexed documents C, the smoothed likelihood of a query term is computed as

$$P(q_i|D, n) = (1 - \alpha_D)P_{ML}(q_i|D, n) + \alpha_D P_{ML}(q_i|C, n) \tag{3}$$

In this equation, α_D is the smoothing factor, and $P(q_i|C, n)$ is the probability that query term q_i was generated by the LM of C given n-gram size n. One of the most common smoothing strategies is Jelinek-Mercer smoothing, which is document-independent and its smoothing factor is $\alpha_D = \lambda$:

$$P(q_i|D, n) = (1 - \lambda)\frac{f_{q_i,D}}{|D|} + \lambda\frac{c_{q_i}}{|C|} \tag{4}$$

where c_{q_i} is the total number of occurrences of q_i in C and $|C|$ is the total number of occurrences of terms in the collection. Dirichlet smoothing is also very popular: this strategy has a smoothing factor $\alpha_D = \frac{\mu}{|D|+\mu}$ that is dependent on the document length:

$$P(q_i|D, n) = \frac{f_{q_i,D} + \mu\frac{c_{q_i}}{|C|}}{|D| + \mu} \tag{5}$$

λ and μ in (4) and (5) are tuning parameters whose values are set empirically.

As mentioned above, $score_{LM}(Q, D, n)$ represents the score of query Q and document D for a given n-gram size n. It is very common that probabilities are computed in logarithmic space since this function is a monotonic transformation that preserves the ranking and avoids precision issues when multiplying probabilities:

$$score_{LM}(Q, D, n) \overset{rank}{=} \log\left(\prod_{i=1}^{n_Q} P(q_i|D, n)\right) = \\ = \sum_{i=1}^{n_Q} \log P(q_i|D, n) \tag{6}$$

In this work, several n-gram sizes are considered in the multigram representation, and the scores obtained for the different indices (i.e. n-gram sizes) using (6) are combined as follows:

$$score_{LM}(Q, D) = \sum_{n=\min_{ngram}}^{\max_{ngram}} \sum_{i=1}^{n_Q} \log P(q_i|D, n) \tag{7}$$

Since document and query transcriptions might have errors, a strategy was proposed in [23] to cope with this issue. It consists in extracting n_{hyp}^{Q} transcription hypotheses $\{Q_{h_1}, \ldots, Q_{h_{n_{hyp}^{Q}}}\}$ for each spoken query Q and computing the query score as

$$score_{LM}(Q, D) = \max_{i \in 1,\ldots,n_{hyp}^{Q}} score_{LM}(Q_{h_i}, D) \tag{8}$$

where Q_{h_i} is the i^{th} transcription of query Q.

The scores retrieved by a QbESDR system are used to decide whether query Q is present in document D or not, so a decision threshold must be established. In order to equalize the distributions of the scores for each query, a common normalization technique for QbESDR is applied in this system, namely the z-norm [56]: 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:

$$score_{LM,z-norm}(Q, D_i) = \frac{score_{LM}(Q, D_i) - \mu_Q}{\sigma_Q} \tag{9}$$

where

$$\mu_Q = \frac{1}{n_m} \sum_{i=1}^{n_m} score(Q, D_i) \tag{10}$$

is the mean of the scores of D_Q and

$$\sigma_Q = \sqrt{\frac{1}{n_m - 1} \sum_{i=1}^{n_m} |score(Q, D_i) - \mu_Q|^2} \tag{11}$$

is the standard deviation of the scores of D_Q. In this way, 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.

3 Proposed approaches for improved LM-based QbESDR

This section describes two approaches to improve the performance of LM-based QbESDR: the first one consists in using n-best transcription hypotheses to compute document probability estimates, and the other aims at introducing positional information using a string alignment strategy.

Table 1 5-best transcription hypotheses of speech utterance "Almost everything"

o d m o s t e v i t e n
o d m o s t e v i t e m
o d m o s t e v i t e i n
o d m o s t e v i t e i
o d m o s t e v i t e i m

3.1 Improved LMs using n-best transcriptions

The key elements of (3) are the probability estimates of term q_i given D and C. This probability estimate is computed as a count of occurrences of q_i divided by the size of the document and the collection, respectively. As mentioned in Section 2, the documents are represented by terms extracted from their 1-best transcription. Automatic phone transcriptions may have errors, and this can lead to incorrect probability estimates. Table 1 shows the 5-best transcription hypotheses of a real speech utterance. It can be observed that there is an agreement about the first eleven phone units, but then the phone sequence is different in all the hypotheses. Obtaining a LM that computes the probability estimates using more than one transcription hypothesis would ease the impact of transcription errors, since it would account for the different alternatives that can be recognized for a given document. Table 2 shows the probability estimates of terms 'm', 'n' and 'o' when using the 1-best transcription in Table 1, and it also shows the probability estimates that would be obtained if the 5-best transcription hypotheses were combined by concatenating them. The probability of term 'n' is equal to that of term 'm' when only the 1-best transcription is considered, but these probability estimates differ significantly when combining the 5-best transcription hypotheses. For term 'o', the probability estimate barely changes regardless the number of hypotheses. This suggests that errors in document transcriptions can be smoothed by combining different transcription hypotheses.

In view of the previous example, the concatenation of different transcription hypotheses to build document LMs is proposed. Formally, given a document D, its n-best transcription hypotheses are extracted from its phone lattice, and then the probability that a term q_i was generated by document D given n-gram size n can be computed as

$$P(q_i|D,n) = \sum_{j=1}^{n_{hyp}^{D}} \frac{f_{q_i,D_{h_j}}}{|D_{h_j}|} \tag{12}$$

where n_{hyp}^{D} is the number of transcription hypotheses that are considered for document D, and D_{h_j} is the j^{th} transcription hypothesis. It is expected that computing $P(q_i|D,n)$ as proposed in (12) will lead to more reliable probability estimates than (2).

Table 2 Example of how probability estimates differ in the example displayed in Table 1 when considering 1-best and 5-best transcription hypotheses

Term	1-best	5-best
'm'	0.083	0.113
'n'	0.083	0.032
'o'	0.167	0.161

3.2 Re-ranking based on positional information

The score computed following (1) depends on the occurrence of the different terms of a query in the document regardless of their order of appearance. It is expected that, when many of the query terms are present in a document (especially those n-grams with larger values of n), the query is most likely to be found in the document, but it is also possible that many terms appear in the document but not in the expected order, leading to false matches. Hence, it is interesting to incorporate positional information to the proposed QbESDR system.

In this work, the use of a string alignment strategy is proposed for taking positional information into account in the QbESDR approach described in Section 2. Specifically, $score_{LM}(Q, D)$ is weighted proportionally to a score $score_{MED}(Q, D)$ given by the MED between the query and document phone transcriptions. MED is defined as the minimum number of editing operations (insertions, deletions and substitutions) that are necessary for transforming one string into another [22]. Therefore, if MED is 0 this score would be 1 and *vice versa*. MED is usually computed following the Wagner-Fischer algorithm [62], which aligns two sequences from beginning to end. Nevertheless, in QbESDR, aligning a short sequence (the query) with a fragment of a longer one (the document) is more suitable. Hence, a modification of this strategy is used, namely subsequence MED (S-MED), which allows the algorithm to skip the initial terms in the document that do not match the sequence of query terms. For this purpose, this modification gives the end position of the alignment, and the start position can be recovered by backtracking of the alignment path [41]. S-MED is inspired in the subsequence DTW algorithm [39] widely used in QbESTD, which is described in detail in Section 4. Given a query $Q = \{q_1, \ldots, q_{n_Q}\}$ and a document $D = \{d_1, \ldots, d_{n_D}\}$, where q_i and d_i are phone 1-grams, $score_{MED}(Q, D)$ can be computed as follows:

$$score_{MED}(Q, D) = \frac{n_Q - S\text{-}MED(Q, D)}{K} \qquad (13)$$

where $S\text{-}MED(Q, D)$ is the minimum edit distance returned by the alignment algorithm, n_Q is the number of query terms and K is the length of the best alignment path. It must be noted that $n_Q - S\text{-}MED(Q, D)$ cannot have negative values because, in the worst case scenario, n_Q editions should be performed to convert a string of length n_Q into a completely different one.

Combining (1) and (13), the new score for query Q and document D is computed as

$$score(Q, D) = score_{LM}(Q, D) \cdot score_{MED}(Q, D) \qquad (14)$$

where $score(Q, D)$, $score_{LM}(Q, D)$ and $score_{MED}(Q, D)$ range from 0 to 1.

The time complexity of the procedure used for computing $score_{MED}(Q, D)$ is $\mathcal{O}(n_Q n_D)$, where n_Q and n_D are the number of terms of the query and the document, respectively. This time complexity is not prohibitive given that n_Q and n_D are not large values in general. Nevertheless, as mentioned in Section 2, n_{hyp}^{Q} transcription hypotheses are searched for each query, so the time complexity would linearly increase with n_{hyp}^{Q}. Hence, instead of computing $score_{MED}(Q, D)$ for all the hypotheses of Q, only the one that led to the greatest score is considered. According to (8):

$$Q^* = \underset{i \in 1, \ldots, n_{hyp}^{Q}}{\arg\max}\, score_{LM}(Q^i, D) \qquad (15)$$

Then, (14) can be rewritten as

$$score(Q, D) = score_{LM}(Q^*, D) \cdot score_{MED}(Q^*, D) \qquad (16)$$

The main purpose of performing string alignment is computing $score_{MED}(Q, D)$ in order to re-rank the documents returned by the probabilistic information retrieval model. Nevertheless, since the computation of this score implies backtracking the alignment path, it is possible to know the initial and final positions of the query-document alignment. Hence, this can also be used to perform QbESTD if the start time and duration of each term are stored in the index.

4 QbESDR using dynamic time warping

DTW is widely used in pattern matching-based approaches for QbESDR and QbESTD, which makes it a suitable reference strategy to evaluate the performance of the techniques proposed in this paper. This section describes in detail the system used in the experiments presented in Section 6, which has three stages: feature extraction, search and score normalization. It must be noted that the proposed system was chosen based on previous work and to enable a straightforward comparison with previous results in [23], but modifications of this system such as using different local and global restrictions on the DTW algorithm [16] can lead to slightly better results. Nevertheless, the analysis of different DTW techniques is out of the scope of this paper.

4.1 Feature extraction

In this system, phone posteriorgrams were used for query and document representation. Given a spoken document and a phone decoder with n_U phone units, the posterior probability of each phone unit is computed for each time frame, leading to a set of vectors of dimension n_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 [61].

4.2 Search algorithm

Let $Q = \{\mathbf{q}_1, \ldots, \mathbf{q}_{F_Q}\}$ and $D = \{\mathbf{d}_1, \ldots, \mathbf{d}_{F_D}\}$ be the phone posteriorgrams of a query and a document with F_Q and F_D frames, where $\mathbf{q}_i$ and $\mathbf{d}_j$ are feature vectors of dimension n_U and $F_Q \ll F_D$. DTW aims at finding the best alignment path between Q and D. Among the different variants of DTW [6, 7, 34, 39, 51], subsequence DTW (S-DTW) was used in this system [39], since it allows alignments between a short sequence (the query) and a longer sequence (the document).

First, a cost matrix $\mathbf{M} \in \Re^{F_Q \times F_D}$ 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

$$M_{i,j} = \begin{cases} c(\mathbf{q}_i, \mathbf{d}_j) & \text{if } i = 1 \\ c(\mathbf{q}_i, \mathbf{d}_j) + M_{i-1,0} & \text{if } i > 1,\ j = 1 \\ c(\mathbf{q}_i, \mathbf{d}_j) + M^*(i, j) & \text{otherwise} \end{cases} \tag{17}$$

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

$$M^*(i, j) = \min\left(M_{i-1,j}, M_{i-1,j-1}, M_{i,j-1}\right) \tag{18}$$

The matrix computed following (17) 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 (18): one step horizontally, one step vertically, or one step horizontally and vertically at the same time. Since DTW aims at minimizing the cost, (18) 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 [18]:

$$cost(\mathbf{q}_i, \mathbf{d}_j) = -\log \frac{\mathbf{q}_i \cdot \mathbf{d}_j}{|\mathbf{q}_i| \cdot |\mathbf{d}_j|} \tag{19}$$

$cost(\mathbf{q}_i, \mathbf{d}_j)$ is normalized in order to turn it into a cost function defined in the interval [0,1] as follows [48]:

$$c(\mathbf{q}_i, \mathbf{d}_j) = \frac{cost(\mathbf{q}_i, \mathbf{d}_j) - cost_{\min}(i)}{cost_{\max}(i) - cost_{\min}(i)} \tag{20}$$

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:

$$b^* = \underset{b \in 1,\ldots,F_D}{\arg\min}\, M_{F_Q,b} \tag{21}$$

Since M is a cumulative matrix cost, each element $M_{F_Q,b}$, $b \in 1, \ldots, F_D$ 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 (21). Then, the first step a^* can be obtained by backtracking the path starting at b^*. This results in an alignment path

$$Path(Q, D) = \{p_1, \ldots, p_k, \ldots, p_K\} \tag{22}$$

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

This system is used for both QbESDR and QbESTD. In the latter task, it is possible that a query appears several times in the same document, so other alignment paths apart from the best one must be considered. In this approach, the top 100 values of $b*$ are considered for each query-document pair as in previous work [26].

4.3 Score normalization

The search stage returns, for each match of a query in a document, the minimum alignment cost M_{F_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 [1]:

$$score(Q, D) = \frac{M_{F_Q,b^*}}{b^* - a^* + F_Q} \tag{23}$$

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 F_Q.

Afterwards, as explained in Section 2, 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 (9) was also applied to the scores.

5 Experimental framework

The experimental frameworks used in this work to assess QbESDR and QbESTD performance were those of MediaEval 2014 Query-by-Example Search on Speech (QUESST 2014) [10] and MediaEval 2013 Spoken Web Search (SWS 2013) evaluations [8], respectively. These databases include 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 3. The audio documents include speech in nine languages (Isixhosa, Isizulu, Sepedi, Setswana, Albanian, Romanian, Basque, Czech and non-native English) in SWS 2013, and in six languages (Albanian, Basque, Czech, non-native English, Romanian, and Slovak) in QUESST 2014. The documents were collected from multiple sources such as broadcast news programs, telephone calls into radio live broadcasts, TED talks or Parliament meetings [9, 11]. Hence, these databases feature 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 aforementioned languages, are of different nature in the two datasets: in SWS 2013, the queries were cut from other recordings, while in QUESST 2014 they were recorded using a mobile phone in order to simulate a regular user querying a retrieval system via speech [9]. There are three different types of queries in QUESST 2014:

- Exact (T1): a hit is produced when an exact match of the lexical representation of the query is found in a document.

Table 3 Summary of the experimental frameworks used in this paper: number of recordings in each set (# recordings); total (Total), minimum (Min) and maximum (Max) duration of the recordings

			Duration		
Dataset	Data	# recordings	Total	Min	Max
SWS 2013	Audio docs	10762	19 h 57 min	0.16 s	145.92 s
	dev queries	505	11 min 26 s	0.17 s	6.35 s
	eval queries	503	11 min 36 s	0.21 s	4.10 s
QUESST 2014	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

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

- 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”.

In SWS 2013, all the queries belong to type T1. Some statistics about the queries are summarized in Table 4.

Two evaluation metrics defined in the experimental protocol of SWS 2013 and QUESST 2014 were used in this work to assess search on speech performance and computational cost.

QbESDR and QbESTD performance are evaluated by means of the maximum term weighted value (MTWV) [17], which is derived from the term weighted value (TWV). Given a system that searches for a set of queries $\mathcal{Q}$ within a set of documents Ω, and given a decision threshold θ for the scores output by the system, 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). Hence, TWV is defined as the complement of the measurement of false alarms and miss detections:

$$TWV(\theta) = 1 - \frac{1}{|\mathcal{Q}|} \sum_{\forall Q \in \mathcal{Q}} \{P_{miss}(Q, \theta) \ + \beta \cdot P_{fa}(Q, \theta)\} \tag{24}$$

where θ is the decision threshold, $P_{miss}(Q, \theta)$ 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:

$$\beta = \frac{C_{fa}(1 - P_{target})}{C_{miss} P_{target}} \tag{25}$$

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).

Table 4 Summary of the queries in the experimental frameworks used in this paper

Dataset	Query type	Query set	# queries	# hits
SWS 2013	T1	dev	505	5589
		eval	503	5562
QUESST 2014	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

Query type denotes the type of queries (All - all queries, 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 of 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):

$$MTWV = TWV(\theta_{opt}) \tag{26}$$

The MTWV was computed using the official evaluation tools of SWS 2013 and QUESST 2014. The values of C_{fa}, C_{miss} and P_{target} were fixed in the evaluation protocols and are equal to 1, 100 and 0.00015, respectively, for SWS 2013 and to 1, 100 and 0.0008, respectively, for QUESST 2014. It must be noted that, in the SWS 2013 evaluation framework, since the task consists in finding the exact position of the queries in the documents, the time interval where the query was detected must overlap the actual position of the hit by at least 50%. In addition to this performance measure, detection error trade-off (DET) curves (plot of the false alarm and miss probabilities at different decision thresholds) are used to present the QbESDR and QbESTD results graphically at different operating points.

The computational cost is measured by means of the searching speed factor [47]:

$$SSF(\mathcal{Q}, \Omega) = \frac{T_{Searching}}{T_{\mathcal{Q}} \cdot T_{\Omega}} \tag{27}$$

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 on an AMD Ryzen 7 1700X @ 3.4 GHz, 8cores/16threads, 32GB RAM using a single thread.

In this work, Lucene[1] was used for indexing and search. It is worth mentioning that the practical implementation of (12) was done by indexing a concatenation of the different transcription hypotheses separated by a non-valid term (i.e. a term that will never be present in any query). For n-grams with $n > 1$, this results in some additional terms that can slightly change the total term counts of the document.

6 Experimental results

This section describes the experimental results obtained in a series of experiments. First, Dirichlet and Jelinek-Mercer smoothing strategies of the probabilistic retrieval model are evaluated. Afterwards, the improvements presented in Section 3 are assessed. Finally, these strategies are compared with DTW-based systems for QbESDR and QbESTD in terms of performance and search time.

A phone decoder was employed to obtain the phone transcriptions used in these experiments. Specifically, the phone decoder developed by the Brno University of Technology (BUT) [53] was used, since it is widely used in the search on speech task, and its public availability allows the reproducibility of the results presented in this paper. The decoder has a hybrid HMM/DNN architecture that uses temporal patterns (TRAPs) for feature representation, leading to speech frames of 25 ms extracted every 10 ms, as described in detail in [53]. Otherwise stated, Czech (CZ) models were used for decoding, since they exhibited better results in previous work [23], although some experiments were also run using the Hungarian (HU) decoder in order to evaluate whether the improvements achieved with the proposed strategies are consistent when using different phone decoders. Both models,

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

provided with the decoding toolkit, were trained on SpeechDat-E databases.[2,3] This led to models of 45 and 61 phone units for CZ and HU, respectively, since those are the units present in the training databases.

The aforementioned models for phone decoding include several silence and noise units to model sounds other than phone units. In these experiments, these silence/noise units where combined into a single unit. Silence/noise occurrences were removed from the queries, since they mostly occurred at the beginning and end of these utterances, so their presence was negligible. Nevertheless, they were kept in the documents, since they are helpful for splitting the documents into sentences so, in this case, these units help to avoid mixing phones from different sentences within the same phone n-gram. The phone decoder is also used to obtain the phone posteriorgrams used in the DTW-system: in this case, 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 [48].

6.1 Tuning of system parameters

The system proposed in Section 2 has several tuning parameters, namely the number of query hypotheses n_{hyp}^{Q}, and the minimum and maximum size of the n-grams min_{ngram} and max_{ngram}. The values tuned for the VSM approach proposed in [23] were adopted in this work to ease system tuning. Specifically, $min_{ngram} = 1$, $max_{ngram} = 5$ and $n_{hyp}^{Q} = 150$.

First, the behavior of the LM retrieval approach when using Dirichlet and Jelinek-Mercer smoothing strategies was evaluated on the dev queries of QUESST 2014 and SWS 2013 datasets. As shown in Fig. 2, the best results were achieved with the Jelinek Mercer smoothing strategy ($\lambda = 0.1$) in both experimental frameworks. It can be noted that, contrarily to the results for text retrieval with long queries reported in [67], the optimal smoothing parameter of both strategies is rather low. This is due to a smaller probability of having unseen terms (i.e. terms with $P(q_i|D) = 0$). Nevertheless, applying smoothing is necessary: in fact, running the same experiment without smoothing (i.e. $\lambda = 0$) led to an MTWV of 0.0518 for QUESST 2014 dev queries.

6.2 Evaluation of the proposed approaches

After selecting the most suitable smoothing function, the proposed strategies for improved LM-based QbESDR were assessed. For this purpose, five different systems were compared:

- baseline: the basic VSM-based strategy presented in [23].
- LM: LM-based system with no improvements (i.e. the system described in Section 2).
- MED: LM system combined with the MED-based re-ranking strategy.
- n-best: LM system with improved LMs using several document transcription hypotheses.
- n-best+MED: LM system featuring both improvements.

The *MED* system has no tuning parameters but, in the case of the *n-best* system, the number of document hypotheses had to be tuned. Hence, experiments using different numbers of phone transcription hypotheses of the documents were carried out. Figure 3 shows

[2] http://www.fee.vutbr.cz/SPEECHDAT-E/sample/czech.html

[3] http://www.fee.vutbr.cz/SPEECHDAT-E/sample/hungarian.html

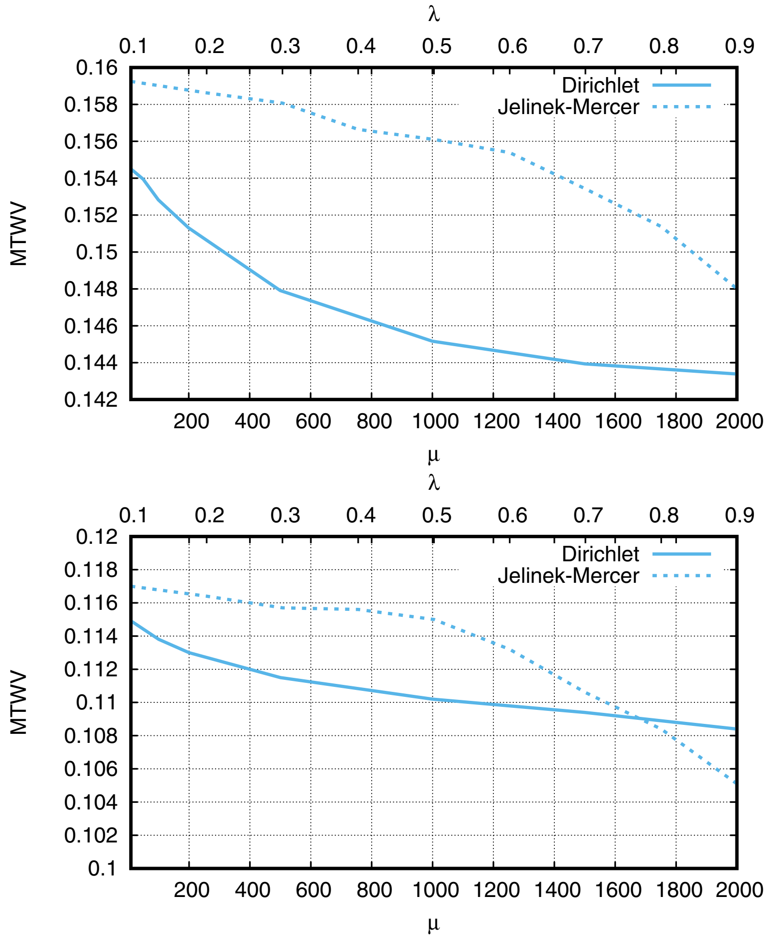


Fig. 2 MTWV obtained with Jelinek-Mercer and Dirichlet smoothing strategies dependent on parameters λ and μ, respectively. These results were computed on the dev queries of QUESST 2014 (top) and SWS 2013 (bottom) using the CZ phone decoder

that, in the dev experiment of QUESST 2014, the best performance is achieved with 200 document hypotheses, and adding more hypotheses leads to a degradation in system performance for all types of queries. The best results for SWS 2013 were achieved using 300 document hypotheses, but the experiment was not run with more hypotheses since the difference in performance achieved by increasing n^D_{hyp} is too small. Hence, from now on, 200 document hypotheses were used in all the experiments.

After parameter tuning, the validity of the proposed strategies was assessed on the eval queries of QUESST 2014 and SWS 2013. In these experiments, only the top 1000 documents retrieved for each query are considered for scoring, as it is commonly done in information retrieval strategies where a ranking of the documents is produced. In this way, when using the *MED* strategy, the number of string alignments is limited to 1000 per query: this avoids re-ranking documents with low scores and, therefore, increasing the efficiency of this strategy.

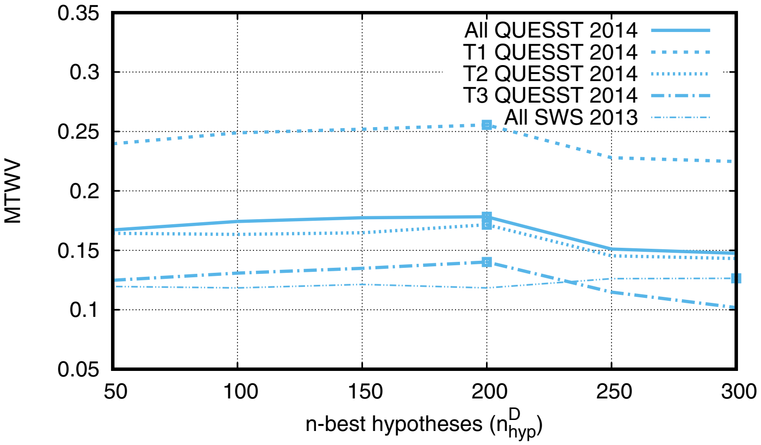


Fig. 3 MTWV for All, T1, T2 and T3 queries of QUESST 2014 dependent on the number of phone transcription hypotheses used in indexing. These results were computed on the dev queries of QUESST 2014 using the CZ phone decoder using Jelinek-Mercer smoothing with $\lambda = 0.1$

The MTWV of the five systems, using CZ and HU decoders, are presented in Table 5. This table shows that, for all the experiments in the two datasets, the three improved strategies outperformed the *baseline* and *LM* results. Compared to the *LM* system, the *n-best+MED* approach led to relative improvements between 34% and 54% (0.03 and 0.06 absolute) depending on the dataset and the phone decoder. The DET curves presented in Fig. 4 further validate the results displayed in Table 5 since they show that the proposed approaches outperform the *baseline* and *LM* systems at almost all the operating points.

Table 5 MTWV for All, T1, T2 and T3 eval queries of QUESST 2014 when using the *baseline*, *LM*, *n-best*, *MED* and *n-best+MED* systems

		QUESST 2014				SWS2013
Decoder	System	All	T1	T2	T3	T1
CZ	Baseline	0.1723	0.2569	0.1325	0.1448	0.0917
	LM	0.1697	0.2518	0.1349	0.1473	0.0904
	n-best	0.1963†	0.2889†	0.1571†	0.1648†	0.1017
	MED	0.1937†	0.2888†	0.1647†	0.1507	0.1139†
	n-best+MED	**0.2314**†	**0.3386**†	**0.1888**†	**0.1753**†	**0.1324**†
HU	Baseline	0.1126	0.1593	0.0871	0.1154	0.0722
	LM	0.1140	0.1656	0.0898	0.1136	0.0725
	n-best	0.1365†	0.1924†	0.1047†	0.1275	0.0810
	MED	0.1414†	0.2061†	0.1156†	0.1258	0.0919†
	n-best+MED	**0.1654**†	**0.2384**†	**0.1255**†	**0.1417**†	**0.1110**†

These results where computed using the CZ and HU phone decoders. LM-based systems feature Jelinek-Mercer smoothing with $\lambda = 0.1$ and $n_{hyp}^{D} = 200$. Results with superindex † show a statistically significant improvement over the baseline system. Statistical significance was computed based on a t-test ($p < 0.05$)

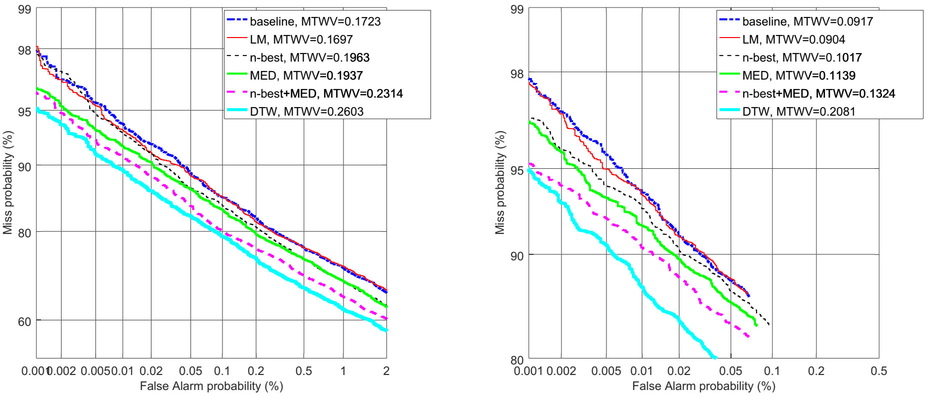


Fig. 4 DET curves obtained from the eval queries of QUESST 2014 (left) and SWS 2013 (right) with the *LM*, *n-best*, *MED* and *n-best+MED* strategies using the CZ phone decoder. The DET curve of a DTW-based system is shown for comparison

6.3 Comparison with a DTW-based approach

Table 6 shows a comparison of the *n-best+MED* system with another based on DTW search on phone posteriorgrams. The table shows that, on the eval experiment of QUESST 2014 with the CZ decoder, the difference between the LM and DTW systems is not statistically significant. DTW outperforms the *n-best+MED* strategy for queries of type T1, there is no statistically significant difference for queries of type T2, and the *n-best+MED* system achieves a clearly better performance for queries of type T3. This behavior is similar for HU decoder but, in this case, there is a statistically significant difference between DTW and *n-best+MED* for All queries.

For SWS 2013 dataset, the performance of the *n-best+MED* strategy is still far from that achieved with DTW. This result was expected since all the queries in this dataset are of type T1, and DTW was also superior to *n-best+MED* for this type of queries in QUESST 2014. DTW performance in SWS 2013 is worse than that achieved for T1 queries in QUESST 2014, which is probably due to the fact that the queries are, in general, much shorter than in QUESST 2014, as suggested by Table 3 (average phone duration of the queries are 7.82

Table 6 MTWV for All, T1, T2 and T3 eval queries of QUESST 2014 and SWS 2013 when using the *n-best+MED* system and a DTW-based strategy

		QUESST 2014				SWS2013
Decoder	System	All	T1	T2	T3	T1
CZ	DTW	0.2603	0.4344†	0.1905	0.0674	0.2081
	LM, n-best+MED	0.2314	0.3386	0.1888	0.1753†	0.1324
HU	DTW	0.2260†	0.3749†	0.1691	0.0541	0.2479
	LM, n-best+MED	0.1654	0.2384	0.1255	0.1417†	0.1110

These results were computed using the CZ and HU phone decoders. *n-best+MED* features Jelinek-Mercer smoothing with $\lambda = 0.1$ and $n_{hyp}^{D} = 200$. Results with superindex † show a statistically significant difference between DTW and *n-best+MED* systems. Statistical significance was computed based on a t-test ($p < 0.05$)

Table 7 Searching speed factor (SSF) of *LM*, *n-best*, *MED* and *n-best+MED* strategies

System	SSF
LM	$6.40 \cdot 10^{-6}$
n-best	$6.34 \cdot 10^{-6}$
MED	$7.25 \cdot 10^{-6}$
n-best+MED	$1.59 \cdot 10^{-5}$
DTW	$4.00 \cdot 10^{-2}$

The SSF of a DTW-based system is shown for comparison. All the search times were measured on the eval experiment of QUESST 2014

and 13.68 phones in SWS2013 and QUESST 2014, respectively). In addition, since in SWS 2013 the queries are cut from longer recordings, they do not usually include silence frames at the beginning and the end of the queries, which sometimes causes the deletion of the first and/or last phones.

The DET plots in Fig. 4 show that, in general, the performance of DTW is superior to that of the *n-best+MED* system, but this difference is small in QUESST 2014 dataset, where both systems exhibit almost the same performance for some operating points.

The SSF of the DTW-based system and the proposed approaches is displayed in Table 7. The *n-best+MED* strategy, which was the top-performing of the proposed ones, is slightly slower than *LM*, *MED* and *n-best*, as expected. Nevertheless, its SSF is three orders of magnitude smaller than that of the DTW-based system. This means that the *n-best+MED* approach for LM-based QbESDR achieved a performance that is not significantly different from that of the DTW-based system, and its search time is reduced to a great extent.

7 Conclusions and future work

This paper presented an approach for QbESDR based on probabilistic retrieval models. In this system, the documents were represented by means of language models, and the query likelihood retrieval model was used to obtain a score for each query-document pair. In addition, two strategies were presented in this paper to enhance the performance of the LM-based probabilistic strategy. Multiple transcription hypotheses for each document were used to obtain improved LMs for document representation. In addition, since the information retrieval model used in this system does not take positional information into account, a re-ranking of the retrieved documents was done by penalizing the scores in function of the minimum edit distance obtained by automatically aligning the query and document phone transcriptions. The latter approach allowed the implementation of a QbESTD system, since the query-document alignment makes it possible to retrieve the start and end times of a match within a document.

The proposed strategy for QbESDR and QbESTD was assessed in the framework of MediaEval 2013 Spoken Web Search (SWS 2013) and MediaEval 2014 Query-by-Example Search on Speech (QUESST 2014) evaluations, and the experimental validation showed that the proposed approaches for enhanced document language models and for incorporating positional information led to a huge improvement in performance in both tasks. In addition, the performance achieved in QUESST 2014 for queries with word reorderings was superior to that of a state-of-art DTW-based system, and there was not a statistically significant difference when dealing with queries with lexical variations. In addition, the search time of the proposed approach, compared to that of the DTW-based strategy, was smaller by several orders of magnitude. The performance exhibited in SWS 2013 dataset was not so close to that of the DTW-based approach: this is coherent with the results exhibited for exact queries in QUESST 2014, as all the queries are of this type in SWS 2013. In general, performance with all the assessed systems was poorer in SWS 2013, probably because the queries were significantly shorter in this experimental framework. Hence, strategies to improve the results for short queries will be explored in the future.

The strategy proposed in this paper to incorporate positional information consists in re-ranking the documents according to their minimum edit distance. It is also possible to incorporate positional information in the language model as proposed in [28], where higher scores are given to those documents where the matched query terms occur close to each other. The idea behind positional language models consists in considering that a term at a position can propagate its occurrence to other nearby positions according to a given probability density function. This implies computing the propagation of each query term occurrence to nearby terms in the documents, which increases the query processing time to a great extent. In future work, efficient strategies to apply this idea to the QbESDR task will be investigated.

In the system described in this paper, the language model of the document collection is obtained from automatic phone transcriptions of all these documents, as usually done in information retrieval. The experimental frameworks used in this work are multilingual, which leads to consider the possibility of using language-dependent collection models. Hence, in future work, automatic techniques for incorporating language information to the proposed QbESDR system will be assessed in order to obtain more suitable language models and to reduce the document space in the search stage.

Acknowledgements This work has received financial support from projects RTI2018-093336-B-C22 (Ministerio de Ciencia, Innovación y Universidades and European Regional Development Fund – ERDF), GPC ED431B 2019/03 (Xunta de Galicia and ERDF) and accreditation ED431G/01 (Xunta de Galicia and ERDF).

References

1. Abad A, Astudillo R, Trancoso I (2013) The L2F spoken web search system for Mediaeval 2013. In: Proceedings of the MediaEval 2013 workshop
2. Abad A, Rodriguez-Fuentes L, Penagarikano M, Varona A, Bordel G (2013) On the calibration and fusion of heterogeneous spoken term detection systems. In: Proceedings of Interspeech, pp 20–24
3. Akiba T, Nishizaki H, Nanjo H, Jones G (2014) Overview of the NTCIR-11 SpokenQuery&Doc task. In: Proceedings of the 11th NTCIR conference, pp 350–364
4. Akiba T, Nishizaki H, Nanjo H, Jones G (2016) Overview of the NTCIR-12 SpokenQuery&Doc-2 task. In: Proceedings of the 12th NTCIR conference on evaluation of information access technologies, pp 167–179
5. Anguera X (2012) Speaker independent discriminant feature extraction for acoustic pattern-matching. In: Proceedings of the 37th international conference on acoustics, speech and signal processing (ICASSP), pp 485–488
6. Anguera X (2013) Information retrieval-based dynamic time warping. In: INTERSPEECH, pp 1–5
7. Anguera X, Ferrarons M (2013) Memory efficient subsequence DTW for query-by-example spoken term detection. In: Proceedings of IEEE international conference on multimedia and expo (ICME), pp 1–6
8. Anguera X, Metze F, Buzo A, Szöke I, Rodriguez-fuentes L (2013) The spoken web search task. In: Proceedings of the MediaEval 2013 workshop
9. Anguera X, Rodriguez-Fuentes L, Buzo A, Metze F, Szöke I, Penagarikano M (2015) 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), pp 5833–5837

10. Anguera X, Rodriguez-Fuentes L, Szöke I, Buzo A, Metze F (2014) Query by example search on speech at Mediaeval 2014. In: Proceedings of the MediaEval 2014 workshop
11. Anguera X, Rodriguez-Fuentes L, Szöke I, Buzo A, Metze F, Penagarikano M (2014) Query-by-example spoken term detection evaluation on low-resource languages. In: Proceedings of spoken language technologies for under-resourced languages workshop (SLTU), pp 24–31
12. Calvo M, Giménez M, Hurtado L, Sanchis E, Gomez J (2014) ELIRF at MediaEval 2014: query by example search on speech task (QUESST). In: Proceedings of the MediaEval 2014 workshop
13. Can D, Saraclar M (2011) Lattice indexing for spoken term detection. IEEE Transactions on Audio, Speech &, Language Processing 19(8):2338–2347
14. Chia T, Li H, Ng H (2007) 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, pp 810–818
15. Chiu J, Wang Y, Trmal J, Povey D, Chen G, Rudnicky A (2014) Combination of FST and CN search in spoken term detection. In: Interspeech, pp 2784–2788
16. Dumpala SH, Raju Alluri KNRK, Gangashetty SV, Vuppala AK (2015) Analysis of constraints on segmental DTW for the task of query-by-example spoken term detection. In: 2015 annual IEEE India conference (INDICON)
17. Fiscus J, Ajot J, Garofolo J, Doddington G (2007) Results of the 2006 spoken term detection evaluation. In: Proceedings of the ACM SIGIR workshop searching spontaneous conversational speech, pp 51–56
18. Gündoğdu B, Saraçlar M (2017) Distance metric learning for posteriorgram based keyword search. In: Proceedings of the 42nd international conference on acoustics, speech and signal processing (ICASSP), pp 5660–5664
19. Hou J, Pham V, Leung CC, Wang L, Xu H, Lv H, Xie L, Fu Z, Ni C, Xiao X, Chen H, Zhang S, Sun S, Yuan Y, Li P, Nwe T, Sivadas S, Ma B, Chng E, Li H (2015) The NNI query-by-example system for MediaEval 2015. In: Proceedings of the MediaEval 2015 workshop
20. Jansen A, Van Durme B, Clark P (2012) The JHU-HLTCOE spoken web search system for MediaEval 2012. In: Proceedings of the MediaEval 2012 workshop
21. Joder C, Weninger F, Wölmer M, Schuller B (2012) The TUM cumulative DTW approach for the Mediaeval 2012 spoken web search task. In: Proceedings of the MediaEval 2012 workshop
22. Jurafsky D, Martin J (2008) Speech and language processing. Prentice Hall, Englewood Cliffs
23. Lopez-Otero P, Barreiro ParaparAJ (2019) Efficient query-by-example spoken document retrieval combining phone multigram representation and dynamic time warping. Inf Process Manag 56:43–60
24. Lopez-Otero P, Docio-Fernandez L, Garcia-Mateo C (2015) GTM-UVIgo systems for the query-by-example search on speech task at MediaEval 2015. In: Proceedings of the MediaEval 2015 workshop
25. Lopez-Otero P, Docio-Fernandez L, Garcia-Mateo C (2015) Phonetic unit selection for cross-lingual query-by-example spoken term detection. In: Proceedings of IEEE automatic speech recognition and understanding workshop, pp 223–229
26. Lopez-Otero P, Docio-Fernandez L, Garcia-Mateo C (2016) Finding relevant features for zero-resource query-by-example search on speech. Speech Comm 84(Supplement C):24–35
27. Lopez-Otero P, Docio-Fernandez L, Garcia-Mateo C (2016) GTM-UVIgo systems for Albayzin 2016 search on speech evaluation. In: Iberspeech 2016, pp 65–74
28. Lv Y, Zhai C (2009) Positional language models for information retrieval. In: Proceedings of ACM SIGIR, pp 299–306
29. Madhavi M, Patil H (2017) Partial matching and search space reduction for qbe-STD. Computer Speech & Language 45:58–82
30. Madhavi M, Patil H (2017) VTLN-warped Gaussian posteriogram for QbE-STD. In: Proceedings of 23rd European signal processing conference (EUSIPCO), pp 563–567
31. Madhavi M, Patil H (2018) Design of mixture of GMMs for query-by-example spoken term detection. Computer Speech & Language (in press)
32. Mangu L, Soltau H, Kuo HK, Kingsbury B, Saon G (2013) Exploiting diversity for spoken term detection. In: Proceedings of the 37th international conference on acoustics, speech and signal processing (ICASSP), pp 8282–8286
33. Manning C, Raghavan P, Schütze H (2008) Introduction to information retrieval. Cambridge University Press, Cambridge
34. Mantena G, Achanta S, Prahallad K (2014) 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):944–953
35. Mantena G, Prahallad K (2014) 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), pp 7128–7132

36. Martinez M, Lopez-Otero P, Varela R, Cardenal-Lopez A, Docio-Fernandez L, Garcia-Mateo C (2014) GTM-UVIgo systems for Albayzin 2014 search on speech evaluation. In: Iberspeech 2014: VIII Jornadas en Tecnología del Habla and IV SLTech Workshop
37. Metze F, Barnard E, Davel M, Heerden CV, Anguera X, Gravier G, Rajput N (2012) The spoken web search task. In: Proceedings of the MediaEval 2012 workshop
38. Metze F, Rajput N, Anguera X, Davel M, Gravier G, Heerden CV, Mantena G, Muscariello A, Pradhallad K, Szöke I, Tejedor J (2012) The spoken web search task at MediaEval 2011. In: Proceedings of the 37th international conference on acoustics, speech and signal processing (ICASSP), pp 5165–5168
39. Müller M (2007) Information retrieval for music and motion. Springer, Berlin
40. Nakagawa S, Iwami K, Fujii Y, Yamamoto K (2013) A robust/fast spoken term detection method based on a syllable n-gram index with a distance metric. Speech Comm 55(3):470–485
41. Navarro G (2001) A guided tour to approximate string matching. ACM Comput Surv 33:31–88
42. Ng K, Zue VW (2000) Subword-based approaches for spoken document retrieval. Speech Comm 32(3):157–186
43. Norouzian A, Rose R (2014) An approach for efficient open vocabulary spoken term detection. Speech Comm 47:50–62
44. Ponte J, Croft W (1998) A language modeling approach to information retrieval. In: Proceedings of ACM SIGIR, pp 275–281
45. Proença J, Castela L, Perdigão F (2015) The SPL-IT-UC query by example search on speech system for MediaEval 2015. In: Proceedings of the MediaEval 2015 workshop
46. Robertson S, Walker S, Jones S, Hancock-Beaulieu MM, Gatford M (1995) Okapi at trec–3. In: Overview of the third text retrieval conference (TREC–3), pp 109–126
47. Rodriguez-Fuentes L, Penagarikano M (2013) MediaEval 2013 spoken web search task: system performance measures. Tech. rep., Software Technologies Working Group, University of the Basque Country, http://gtts.ehu.es/gtts/NT/fulltext/rodriguezmediaeval13.pdf
48. Rodriguez-Fuentes L, Varona A, Penagarikano M (2014) GTTS-EHU systems for QUESST at MediaEval 2014. In: Proceedings of the MediaEval 2014 workshop
49. Rodriguez-Fuentes L, Varona A, Penagarikano M, Bordel G, Diez M (2014) 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), pp 7869–7873
50. Sakamoto N, Yamamoto K, Nakagawa S (2014) 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, pp 419–424
51. Sakoe H, Chiba S (1978) Dynamic programming algorithm optimization for spoken word recognition. IEEE Transactions on Acoustics, Speech and Signal Processing 23(1):43–49
52. Salton G, Wong A, Yang C (1975) A vector space model for automatic indexing. Commun ACM 18(11):613–620
53. Schwarz P (2009) Phoneme recognition based on long temporal context. PhD thesis, Brno University of Technology
54. Spärck Jones K, Walker S, Robertson S (2000) A probabilistic model of information retrieval: development and comparative experiments. Information Processing & Management 36(6):809–840
55. Szöke I, Burget L, Grézl F, Ondel L (2013) BUT SWS 2013 - massive parallel approach. In: Proceedings of the MediaEval 2013 workshop
56. Szöke I, Burget L, Grézl F, Černocký J, Ondel L (2014) 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), pp 7899–7903
57. Szöke I, Rodriguez-Fuentes L, Buzo A, Anguera X, Metze F, Proença J, Lojka M, Xiong X (2015) Query By example search on speech at MediaEval 2015. In: Proceedings of the MediaEval 2015 workshop
58. Tejedor J, Toledano D (2016) The ALBAYZIN 2016 search on speech evaluation plan. https://iberspeech2016.inesc-id.pt/wp-content/uploads/2016/06/EvaluationPlanSearchonSpeech.pdf last Accessed 9 Jan 2018
59. Tejedor J, Toledano D, Anguera X, Varona A, Hurtado L, Miguel A, Colás J (2013) Query-by-example spoken term detection ALBAYZIN 2012 evaluation: overview, systems, results, and discussion. EURASIP Journal on Audio, Speech, and Music Processing 2013(23)
60. Tejedor J, Toledano D, Lopez-Otero P, Docio-Fernandez L, Garcia-Mateo C (2016) Comparison of ALBAYZIN query-by-example spoken term detection 2012 and 2014 evaluations. EURASIP Journal on Audio, Speech, and Music Processing 2016(1)
61. Varona A, Penagarikano M, Rodriguez-Fuentes L, Bordel G (2011) On the use of lattices of time-synchronous cross-decoder phone co-occurrences in a SVM-phonotactic language recognition system. In: INTERSPEECH, pp 2901–2904
62. Wagner RA, Fischer MJ (1974) The string-to-string correction problem. J ACM 21(1):168–173

63. Witten IH, Moffat A, Bell TC (1999) Managing gigabytes, 2nd edn. Compressing and Indexing Documents and Images. Morgan Kaufmann Publishers Inc., San Francisco
64. Xu H, Hou J, Xiao X, Pham V, Leung CC, Wang L, Do V, Lv H, Xie L, Ma B, Chng E, Li H (2016) 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), pp 6030–6034
65. Xu H, Yang P, Xiao X, Xie L, Leung CC, Chen H, Yu J, Lv H, Wang L, Leow S, Ma B, Chng E, Li H (2015) 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), pp 5191–5195
66. Yang P, Xu H, Xiao X, Xie L, Leung CC, Chen H, Yu J, Lv H, Wang L, Leow S, Ma B, Chng E, Li H (2014) The NNI query-by-example system for MediaEval 2014. In: Proceedings of the MediaEval 2014 workshop
67. Zhai C, Lafferty J (2001) A study of smoothing methods for language models applied to ad hoc information retrieval. In: Proceedings of ACM SIGIR, pp 268–276
68. Zhang Y, Glass J (2009) Unsupervised spoken keyword spotting via segmental DTW on Gaussian posteriorgrams. In: IEEE automatic speech recognition and understanding workshop (ASRU), pp 398–403

Paula Lopez-Otero is an associate researcher at the University of A Coruña. She obtained her degree in Telecommunication Engineering and the P.hD. degree from the University of Vigo. Her research interests cover speech processing topics such as search on speech, speaker de-identification and emotional state detection from speech. She has published more than 40 journal and conference papers, and she has been reviewer of multiple conferences (Interspeech, ICASSP, Speaker Odyssey, among others) and journals (IEEE TCYB, IEEE TAFFC, CSL, among others).

Javier Parapar is an Assistant Professor at the University of A Coruña (Spain). He was President of the Spanish Society for IR from 2014 to 2018. Javier Parapar holds a B.Sc.+M.Sc. in Computer Science and he got his Ph.D. in Computer Science (cum laude) in 2013, both from the University of A Coruña (Spain). His current research interests include but are not limited to information retrieval evaluation, recommender systems, text mining, and summarization. He regularly serves as reviewer and PC member of conferences such as ACM RecSys, The Web Conference (WWW), ACM SIGIR, ECIR, etc. He is a member of IP&M editorial board and a regular reviewer for journals such as ACM TOIS, IRJ, DKE and TKDE.

Alvaro Barreiro is a Professor in Computer Science at the University of A Coruña (Spain) and the group leader of the Information Retrieval Lab. He has supervised five doctoral theses and five research projects of the National R&D program, as well as other regional projects and R&D projects with companies. His research interests include information retrieval models, efficiency in information retrieval systems, text and data analysis and classification, evaluation and recommender systems. He has been acknowledged as ACM Senior Member and he is a member of ACM and ACM-SIGIR, BCS and BCS-IRSG, AEPIA and SERI societies.