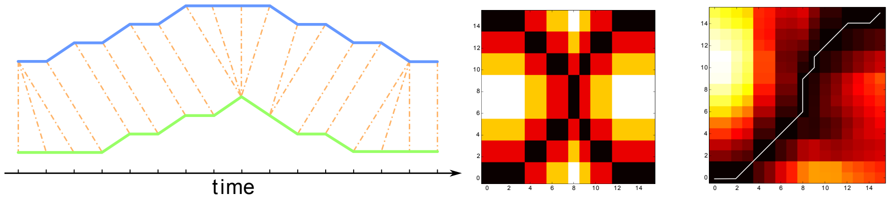
Excel@FIT 2015
http://excel.fit.vutbr.cz

Query-by-Example Spoken Term Detection

Miroslav Skácel*



Abstract
This paper aims at a search in a large speech database with zero or low-resource languages by spoken term example in a spoken utterance. The data can not be recognized by Automatic Speech Recognition system due to a lack of resources. A modern method for searching patterns in speech called Query-by-Example is investigated. This technique exploits a well-known dynamic programming approach named Dynamic Time Warping. An analysis of different distance metrics used during the search is provided. A scoring metric based on normalized cross entropy is described to evaluate the system accuracy.

Keywords: Query-by-Example — Spoken Term Detection — Dynamic Time Warping — acoustic pattern search — unsupervised detection in speech

Supplementary Material: *N/A*

*xskace00@stud.fit.vutbr.cz, *Faculty of Information Technology, Brno University of Technology*

1. Introduction

We propose a system that was created in a collaboration of the author of this paper, Igor Szöke and Lukáš Burget. All are members of speech research group BUT Speech@FIT. The system was used for evaluation in Query by Example Search on Speech Task (QUESST[1]) in MediaEval 2014. In this paper, we focus only on the parts of the system proposed, designed and implemented by the author. Other system parts created by other members are briefly described to present the whole work and more detailed information could be found in the references.

The system was built to deal with searching of a user-specified term in a large speech database where the term is specified in an audio format. In fact, it is impossible to physically listen to hundreds or thousands hours of speech, moreover with no a priori knowledge of the language. The database for testing and evaluation is described in Section 2. In Section 3, we describe a scoring metric to evaluate the system performance. An application of Query-by-Example is investigated in Section 4.

In our work, we exploit classical Dynamic Time Warping algorithm detailed in Section 5 to compare two spoken documents represented as sequences of multidimensional vectors obtained by feature extraction where the one containing the keyword to be found is known as a *query* and the other one to be searched in is referred as an *utterance*.

2. Datasets

The database used for our evaluation of proposed system was originally created for QUESST. Speech data were collected at several institutions. The database consists of about 23 hours of speech in 6 languages. The search utterances were automatically extracted from longer recordings and checked manually for un-

[1] http://www.multimediaeval.org/mediaeval2014/quesst2014/

Language	Utterances (mins/files)	Queries (dev/eval)	Speech type
Albanian	127/968	50/50	read
Basque	192/1841	70/70	broadcast
Czech	237/2652	100/100	conversation
NNEnglish[2]	273/2438	138/138	TEDx
Romanian	244/2272	100/100	read
Slovak	312/2320	102/97	parliament
SUM	1385/12491	560/555	mixed

Table 1. Database description [2]. For each language, the total length in minutes and the number of utterances are shown. The number of development and evaluation queries are balanced. Last, the type of speech is mentioned.

wanted qualities. The queries to be searched were recorded manually. All data have PCM encoding at 8kHz, 16bits/sample and WAV format [1]. The database has one set of utterances for both development and evaluation. The queries are split into two sets for each part of the task. The summary of database can be seen in Table 1.

3. Scoring Metrics

The proposed system was evaluated for its accuracy. The goal was to detect the presence of a query in an utterance regardless the position of the match. The score represents a value of how we are sure about it.

The primary scoring metric is called **normalized cross entropy** cost (C_{nxe}). C_{nxe} measures the fraction of information, with regard to the ground truth, that is not provided by system scores, assuming that they can be interpreted as log-likelihood ratios (**llr**). The best system score is $C_{nxe} \approx 0$ and a non-informative (random) system returns $C_{nxe} = 1$. System scores $C_{nxe} > 1$ indicate severe miscalibration of the log-likelihood ratio scores. C_{nxe} is computed on system scores for a reduced subset of all possible set of trials. Each trial consists of a query q and a segment x. For each trial, the ground truth is a *True* or *False* depending on whether q actually appears in x or not [3].

The cross entropy measures both discrimination between target and non-target trial and calibration. To estimate the calibration loss, a system can be optimally recalibrate using a simple reversible transformation, such as:

$$\hat{\mathbf{l}}\mathbf{lr} = \gamma \cdot \mathbf{llr} + \delta, \tag{1}$$

where γ and δ are calibration parameters that can be used to minimize the normalized cross entropy [4]:

$$C_{nxe}^{min} = \min_{\gamma,\delta}\{C_{nxe}\} \tag{2}$$

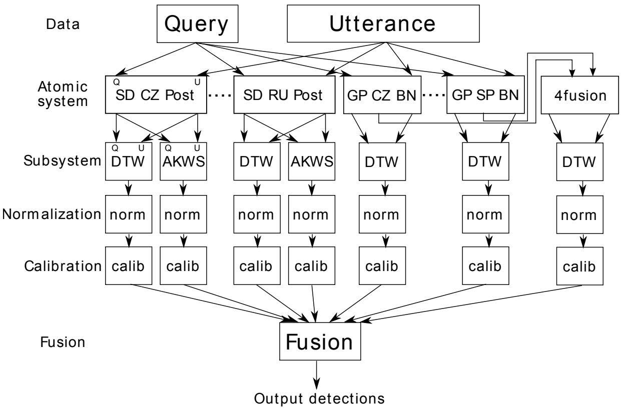


Figure 1. Query-by-Example system. Q means queries as an input, U stands for utterances as an input, SD means SpeechDat atomic systems where the output are phoneme-state posteriors, GP stands for GlobalPhone atomic systems where the output are bottleneck features [2].

4. Query-by-Example System

Query-by-Example (QbE) is a technique to search an example of an object or at least a part of it in some other object. QbE has been used in application like sound classification, music retrieval or spoken document retrieval. As mentioned, a query is an example of an object to be found and in our case, it is the spoken term to search. The spoken term is a word or a word phrase an it is represented as a speech cut. This query is then searched in set of speech utterances and segments similar to searched query are returned. QbE is used when not enough resources for training acoustic models are available. In other words, it is a low-resource technique for pattern search in speech [5].

Our QbE Spoken Term Detection system is based on **phoneme-state posterior** (POST) extractors and **bottleneck** (BN) features extractors (Figure 1) based on artificial neural networks. BUT phoneme recognizer *phnrec*[3] [6] was used for feature extraction. In total, we used 7 best atomic systems according to C_{nxe}. These generated features are then processed by two QbE subsystems. The first one is based on Acoustic Keyword Spotting (AKWS) where we build a Hidden Markov Model (HMM) for each query and then a log-likelihood between the query and a background

[2] non-native English

[3] http://speech.fit.vutbr.cz/cs/software/phoneme-recognizer-based-long-temporal-context

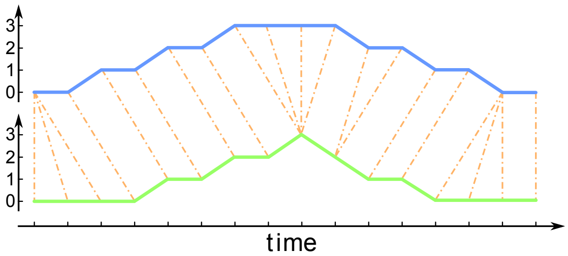


Figure 2. A warping between two different time series. The blue and the green horizontal lines represent two different time series. Each point from one series is optimally aligned with one or more points from the other one and vice versa which allow us to compare even time series with different duration. The warping is shown by the orange dash-dotted vertical lines. Evidently, the warping of series to each other is a non-linear operation [8].

model is calculated (more details in [2]). The second subsystem is based on Dynamic Time Warping and is described in detail in the following section. The output of these subsystems is a set of all detections of given query in the utterance and their score. A score normalization and calibration follow. The score is normalized using the set of obtained detections, one score for each pair query-utterance is produced and the score is then calibrated.

At last, the results of all calibrated subsystems are fused. Both the calibration and the fusion is performed with respect to C_{nxe} (more details in [2]).

The system was built by other team members. The author of this paper was investigating the second subsystem based on Dynamic Time Warping only.

5. Dynamic Time Warping

Dynamic Time Warping (DTW) is a dynamic programming approach used for comparing and finding an optimal alignment between two different time series. These time series are warped in time or in speed. DTW has been originally exploited for comparison of speech patterns in automatic speech recognition system and it has been applied to cope with time-dependent data with time deformations and inconsistencies or different speeds [7]. In Figure 2, a warping between two different time series is shown.

To describe our system more formally, let us consider an utterance $\mathbf{U} = \{\mathbf{u_1}, \ldots, \mathbf{u_N}\}$ as a time-dependent sequence of N vectors and a query $\mathbf{Q} = \{\mathbf{q_1}, \ldots, \mathbf{q_M}\}$ as a time-dependent sequence of M vectors. All vectors $\mathbf{u} \in \mathbf{U}$ and $\mathbf{q} \in \mathbf{Q}$ have the same dimensionality L.

6. Distance Metrics

Different metrics[4] for measuring distances between query-utterance vectors were used. The distance metric to compare two vectors $\mathbf{u}$ and $\mathbf{q}$ is defined in general as $d : \mathbf{u} \times \mathbf{q} \to \mathbb{R}$.

The **cosine distance** d_{cos} is defined as:

$$d_{cos}(\mathbf{u},\mathbf{q}) = 1 - \frac{\mathbf{u} \cdot \mathbf{q}}{|\mathbf{u}| \cdot |\mathbf{q}|}, \tag{3}$$

where $\cdot$ represents the dot product and $|\mathbf{u}|$ stands for the magnitude of vector $\mathbf{u}$. The range of the d_{cos} is given by the interval $[0,2]$ where 0 denotes identical vectors.

The **Pearson product-moment correlation distance** d_{corr} is defined by:

$$d_{corr}(\mathbf{u},\mathbf{q}) = 1 - \frac{(\mathbf{u}-\bar{\mathbf{u}}) \cdot (\mathbf{q}-\bar{\mathbf{q}})}{|(\mathbf{u}-\bar{\mathbf{u}})| \cdot |(\mathbf{q}-\bar{\mathbf{q}})|}, \tag{4}$$

where $\bar{\mathbf{u}}$ represents the mean value of vector $\mathbf{u}$. The range of the d_{corr} distance falls into the interval $[0,2]$ where 0 means identical vectors. Evidently, the only difference between the d_{corr} and the d_{cos} is that the input vectors are mean normalized within the d_{corr}.

The **Euclidean distance** d_{euc} is defined as:

$$d_{euc}(\mathbf{u},\mathbf{q}) = \sqrt{\sum_{k=1}^{L} (u_k - q_k)^2}, \tag{5}$$

where u_k is the k-th element of vector $\mathbf{u}$. The range of the d_{euc} lies in the interval $[0,+\infty)$ where 0 stands for identical vectors.

The **log-likelihood based on the cosine distance** d_{logcos} is defined by [5]:

$$d_{logcos}(\mathbf{u},\mathbf{q}) = -\log\left(\frac{\mathbf{u} \cdot \mathbf{q}}{|\mathbf{u}| \cdot |\mathbf{q}|}\right), \tag{6}$$

where the expression in parentheses is the cosine similarity. The range of the d_{logcos} is given by the interval $[0,+\infty)$ where 0 denotes identical vectors.

The last metric, the **log-likelihood based on the dot product** d_{logdot}, is defined as:

$$d_{logdot}(\mathbf{u},\mathbf{q}) = -\log(\mathbf{u} \cdot \mathbf{q}), \tag{7}$$

where $\cdot$ represents the dot product. The range of the d_{logcos} lies in the interval $[0,+\infty)$ where 0 denotes identical vectors.

In addition to these distance metrics, several others were used in experiments without significant results (e.g. Mahalanobis, Bray-Curtis, Canberra, etc.)

[4] http://docs.scipy.org/doc/scipy-0.14.0/reference/generated/scipy.spatial.distance.cdist.html

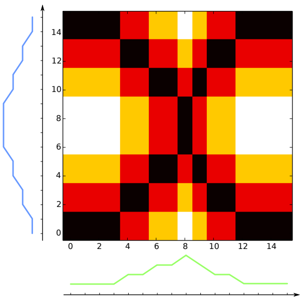


Figure 3. A distance matrix (on the left) for two real-valued sequences from Figure 2. The Euclidean distance (5) was used to measure distances. The darker colors denote areas where given vectors are similar to each other and the lighter colors symbolize regions of a difference. A cumulative matrix (on the right) corresponds to the distance matrix. The white line represents the optimal warping path [8].

By calculating distances between all possible query-utterance vectors $\mathbf{u} \in \mathbf{U}$ and $\mathbf{q} \in \mathbf{Q}$, we obtain a **distance matrix** $\mathbf{D} \in \mathbb{R}^{N \times M}$ where each cell $D(n,m)$ of the matrix is defined by $d(\mathbf{u_n}, \mathbf{q_m})$ [7]. Figure 3 depicts the distance matrix for two real-valued one-dimensional time series (sequences of real numbers) shown in Figure 2.

A **cumulative matrix C** accumulates distance values from a distance matrix. Each cell value depends on its predecessor cells (horizontal, vertical and diagonal). The predecessor cell with the lowest value is taken and accumulated with the current cell. The weight factor w_x was set to 1 for all directions. Formally [7]:

$$C(n,m) = \min \begin{cases} C(n-1,m-1) + w_d \cdot D(n,m) \\ C(n-1,1) + w_h \cdot D(n,m) \\ C(n,m-1) + w_v \cdot D(n,m) \end{cases} \quad (8)$$

Since the query appears anywhere in the utterance, the accumulation starts from the origin point $(0,0)$ of the distance matrix **D** and can reset in the first row (time-dependent axis) of the utterance. Last, the cumulative matrix is normalized by length. In Figure 3, the cumulative matrix is depicted.

A **starting-point matrix S** stores starting points for all possible paths to avoid further exhaustive computation of paths using back-tracking. When searching for paths, the end-point is selected from a distortion profile (the last row of cumulative matrix) and the start-point corresponds to the value from starting-point matrix.

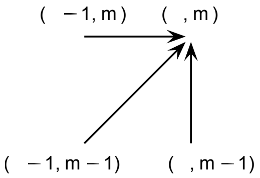


Figure 4. Possible predecessor cells for the cell in coordinates (n,m) lie in a horizontal $(n-1,m)$, a vertical $(n,m-1)$ and a diagonal $(n-1,m-1)$ direction.

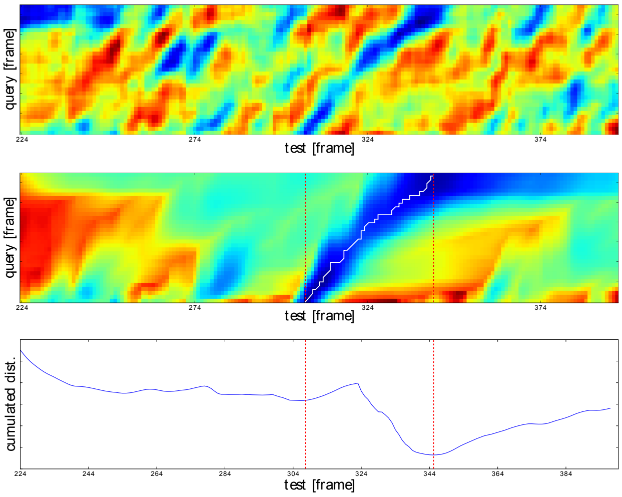


Figure 5. An example of a query match in an utterance. From top to bottom: a distance matrix showing distances between all pairs of query-utterance vectors; a cumulative matrix holding accumulated distances (the optimal warping is shown as the white line); a distortion profile used for best paths selection.

An example of a query match and a description of all presented matrices is shown in Figure 5.

7. Results

In Table 2, a comparison of presented distance metrics on the development dataset is shown. For the posteriors, the best metric was d_{logcos}. The bottlenecks work the best using d_{cos} metric. The most robust metric was considered d_{corr} as it gave us good results regardless the type of the features. The worst distance metric was d_{euc} which does not work well for any type of features. The 7 atomic systems for fusion were selected during experiments and adding extra systems does not improve overall score significantly. The best single system using features from a recognizer trained on Czech language matches Czech and Slovak part of the database which explains its highest accuracy.

8. Conclusions

The proposed system using the fusion outperformed all the other systems in QUESST evaluations in MediaEval 2014. We conclude a superiority of bottleneck features for the fusion. The single best system designed by the author achieved also excellent results and scored the second. The overall evaluation results are shown in Table 3.

Features	d_{corr}	d_{cos}	d_{euc}	d_{logcos}	d_{logdot}
SD CZ POST	0.687	0.768	0.852	**0.649**	0.658
SD HU POST	**0.646**	0.712	0.805	0.679	0.691
SD RU POST	0.653	0.706	0.789	**0.652**	0.662
GP CZ BN	0.593	**0.585**	0.777	0.722	-
GP PO BN	0.659	**0.650**	0.882	0.819	-
GP RU BN	0.668	**0.658**	0.862	0.814	-
GP SP BN	0.673	**0.663**	0.849	0.822	-
GP CZ+PO+RU+SP BN 4fusion	0.586	**0.579**	0.761	0.713	-

Table 2. A comparison of distance metrics for 7 atomic systems and the fusion. The atomic systems used state-phoneme posteriors (POST) from recognizers trained on Czech (CZ), Hungarian (HU) and Russian (RU) languages from SpeechDat (SD) database; and bottlenecks (BN) from Czech (CZ), Portuguese (PO), Russian (RU) and Spanish (SP) languages from GlobalPhone (GP) database. The overall C_{nxe}^{min} score for the development dataset for all types of queries is shown (lower is better). The best distance metric for each system is indicated in bold [2].

System	C_{nxe} / C_{nxe}^{min}
BUT 4fusion	0.473 / 0.466
BUT GP CZ BN	**0.536 / 0.528**
NTU-NPU-I2R	0.602 / 0.598
EHU	0.621 / 0.599
SPL-IT	0.659 / 0.508
CUHK	0.683 / 0.659
IIIT-H	0.921 / 0.812

Table 3. QUESST 2014 results for the evaluation dataset. The best systems for 6 out of 15 registered participants are listed. C_{nxe} and C_{nxe}^{min} score for each system is presented. The winning system was BUT 4fusion system. The single system based on DTW and developed by the author (shown in bold) outperformed other participants according to C_{nxe} metric.

References

[1] Xavier Anguera, Luis Javier Rodriguez-Fuentes, Igor Szöke, Andi Buzo, and Florian Metze. Query by Example Search on Speech at MediaEval 2014. In *Proceedings of the Mediaeval 2014 Workshop*, 2014.

[2] Igor Szöke, Miroslav Skácel, Lukáš Burget, and Jan Černocký. Coping with Channel Mismatch in Query-by-Example - BUT QUESST 2014. *Accepted at ICASSP 2015*, 2015.

[3] Xavier Anguera, Luis Javier Rodriguez-Fuentes, Igor Szöke, Andi Buzo, Florian Metze, and Mikel Penagarikano. Query-by-Example Spoken Term Detection Evaluation on Low-resource Languages. In *Proceedings of the 4th International Workshop on Spoken Language Technologies for Under-resourced Languages SLTU-2014. St. Petersburg, Russia*, pages 24–31. International Speech Communication Association, 2014.

[4] Luis Javier Rodriguez-Fuentes and Mikel Penagarikano. MediaEval 2013 Spoken Web Search Task: System Performance Measures. 2013.

[5] Javier Tejedor, Michal Fapšo, Igor Szöke, Jan "Honza" Černocký, and František Grézl. Comparison of Methods for Language-dependent and Language-independent Query-by-Example Spoken Term Detection. *ACM Trans. Inf. Syst.*, 30(3):18:1–18:34, September 2012.

[6] Petr Schwarz. *Phoneme Recognition Based on Long Temporal Context*. PhD thesis, 2009.

[7] Meinard Müller. *Information Retrieval for Music and Motion*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2007.

[8] Stan Salvador and Philip Chan. Toward Accurate Dynamic Time Warping in Linear Time and Space. *Intell. Data Anal.*, 11(5):561–580, October 2007.