QUERY BY EXAMPLE SEARCH WITH SEGMENTED DYNAMIC TIME WARPING FOR NON-EXACT SPOKEN QUERIES

Jorge Proença, Arlindo Veiga, Fernando Perdigão

Instituto de Telecomunicações, Coimbra, Portugal
Electrical and Computer Eng. Department, University of Coimbra, Portugal

ABSTRACT

This paper presents an approach to the Query-by-Example task of finding spoken queries on speech databases when the intended match may be non-exact or slightly complex. The built system is low-resource as it tries to solve the problem where the language of queries and searched audio is unspecified. Our method is based on a modified Dynamic Time Warping (DTW) algorithm using posteriorgrams and extracting intricate paths to account for special cases of query match such as word re-ordering, lexical variations and filler content. This system was evaluated on the MediaEval 2014 task of Query by Example Search on Speech (QUESST) where the spoken data is from different languages, unknown to the participant. We combined the results of five DTW modifications computed on the output of three phoneme recognizers of different languages. The combination of all systems provided the best performance overall and improved detection of complex case queries.

Index Terms— Query-by-example, audio search, dynamic time warping, pattern matching

1. INTRODUCTION

Searching large databases of audio documents with a small query is as task commonly known as Spoken Term Detection (STD). Typically, it involves a text-based query and a spoken dataset of a single language for which there are a large amount of resources to build Automatic Speech Recognition (ASR) systems, leading to the audio documents being indexed at a word level. Challenges such as the NIST 2006 STD Evaluation [1] and the 2013 Open Keyword Spotting Evaluation [2] have attracted research on the STD task.

Query-by-Example (QbE) is a task that differs from STD in the sense that no textual information is considered and the query must be audio based, leading to the problem of finding audio using audio [3-6]. The necessity for QbE arises from cases where the language is unknown or has few resources, or if multilingual databases are searched. It is expected to match spoken queries to larger audio files and, usually, it involves the detection of unconstrained audio tokens in the data (zero-resources) [5] or the use of phonetic recognizers for other languages (low-resources) with the extraction of features such as posterior probabilities of phonemes [3,4]. Most works use classical techniques such as Dynamic Time Warping (DTW) [3] or Acoustic Keyword Spotting (AKWS) [7]. Systems for QbE search keep improving with recent advances such as combining spectral acoustic and temporal acoustic models [8], combining a high number of subsystems using both AKWS and DTW and using bottleneck features of neural networks as input [9], new distance normalization techniques [10] and several approaches to system fusion and calibration [11].

The MediaEval task of Query by Example Search on Speech (QUESST) [12,13,20] (formerly known as Spoken Web Search) is a suitable benchmark to tackle the QbE problem. The 2014 edition presents some differences to the previous years' challenges by introducing complex query-reference matches. In addition to the exact match of the query to reference (type 1), there are occurrences where a portion of the beginning or the end of the query may not match (type 2) and where the words in searched audio may be in a different order or with small extra content in between (type 3). It is also not exactly a STD problem, since it is only necessary to retrieve the matching document, making it a spoken document retrieval (SDR) problem. The dataset is multilingual and of mixed acoustical conditions and speaking styles, further increasing the challenging aspect of the task.

Our system performs five segmenting strategies of Dynamic Time Warping (DTW) applied to state-level posterior probabilities comparison, and combines their results. These strategies target the match cases defined in the challenge. It also fuses the results of the same approach applied to the output of three phonetic recognizers of three languages. Therefore, the search is based on a phonetic-level match, and no word-level information is acquired.

2. DATASET AND SCORES

The QUESST 2014 dataset [13] includes 23 hours of speech in 6 languages: Albanian, Basque, Czech, non-native English, Romanian and Slovak. Recordings with an 8 kHz sampling rate and average duration of 6.6 seconds were extracted from different sources of larger recordings such as broadcast news, lectures, read speech and conversations. The various languages are randomly distributed in the data, and no information is given to the participant about which language an utterance belongs to, requiring robust unsupervised approaches.

978-0-9928626-3-3/15/$31.00 ©2015 IEEE

Queries were manually recorded in different conditions from the utterances, emulating the use of a retrieval system with speech. Two sets were created (development and evaluation) and three types of queries were defined, that present varying matching conditions to the utterances:

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

The type 2 and 3 queries are the novelty in this edition of the challenge, and require complex approaches. The fact that there are different languages and speaking styles in the data as well as query and utterances conditions varying, contribute to the constraint of building low or zero-resource systems.

The results of system performance will be presented by the scoring metrics of normalized cross entropy cost (Cnxe) and Actual Term Weighted Value (ATWV). Cnxe has been used for speaker/language recognition and evaluates system scores, with no concern for hard yes/no decision [14]. It interprets scores as log-likelihood ratios and measures the amount of information that is not provided by the scores compared to the ground truth where a perfect system would have Cnxe ≈ 0. ATWV evaluates system decision and takes into account not only false alarm and miss error rates, but also a pre-defined false alarm error cost (Cfa=1) and a miss error cost (Cmiss=100), as well as a prior of the target trials (prior probability of finding a query in an audio file, Pt=0.0008).

3. SYSTEM DESCRIPTION

3.1. Phonetic Recognizer

The initial step was to run unconstrained phonetic recognition on all audio and extract frame-wise posterior probability of phonemes. An early idea was to employ a phoneme recognizer based on Hidden Markov Models and a keyword spotting system such as our in-house one [15], but no easy generation of posteriorgrams could be obtained. Therefore, we decided to use an external tool, a phoneme recognizer from Brno University of Technology [16] based on long temporal context and neural network classifiers. Three systems were available for 8 kHz audio, based on three languages trained with SpeechDat-E databases [17]: Czech, Hungarian and Russian. This makes Czech the only recognizer matching a language of the QUESST 2014 database. The use of different languages leads to using separate sets of phonemes, and hopefully fusing the results will describe the similarities of a query to the searched audio in an improved manner.

All the queries and audio files were run through the three systems, and the state-level posteriorgrams were extracted. This means that, per frame, there are values for the three states of each phoneme, although the state sequence is often well defined. Leading and trailing silences or noises were cut on queries, from the initial and final frames that had a high probability of corresponding to either silence or noise (considered if the sum of the 3 states of ‘int’, ‘pau’ and ‘spk’ phones was greater than 50% for the average of the 3 languages).

3.2. Dynamic Time Warping

The posteriorgrams of a query and searched audio can be compared frame-wise with a local distance matrix where Dynamic Time Warping (DTW) can be applied. We implemented a version of the DTW approach for the proposed task, which will be modified in ways described on the next subsection. The basis is defined here: as in [3], the local distance is obtained from the dot product of posterior probability vectors of query $\vec{q}$ (with N frames) and audio $\vec{x}$ (with M frames) for each frame pairing:

$$D(\vec{q},\vec{x}) = -\log(\vec{q}.\vec{x}) \tag{1}$$

However, the posteriorgram distributions $\vec{q}$ and $\vec{x}$ are smoothed with a back-off with $\lambda = 10^{-4}$, to assure that the dot product is not zero:

$$\vec{q}' = (1-\lambda)\vec{q} + \lambda\vec{u} \tag{2}$$

Here $\vec{u}$ is a uniform distribution of probability for each posteriorgram state. The final result is a local distance matrix of size $N \times M$ where DTW is applied.

The start and end of a DTW path was not restricted in the searched audio, so that the query match can happen at any location. As for local path restrictions, we tested a small number of alternative options, but the most versatile was found to be allowing a path to continue in 3 directions in the distance matrix to directly adjacent points with the lowest local distance: horizontally, vertically and diagonally as shown in Figure 1. All these movements have equal (unitary) weight, meaning that local distance values are simply summed along a path.

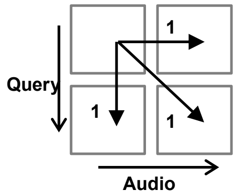

Fig. 1. Schematic of possible unitary weighted movements considered for path creation in Dynamic Time Warping.

The final path distance is normalized by the length of the path. This is the basic approach (named A1) and outputs the lowest normalized distance found (from the best path). It is the basis from which the following approaches will be constructed.

3.3. Modifications on the DTW

The special query types include lexical variations at the beginning or end, word reordering or filler content. To tackle these types, we developed four additional approaches based on changing the DTW method to get different segments or, more precisely, allowing different behaviors for DTW paths. During the DTW algorithm we keep a matrix of accumulated distances of the best path for each point as well as a matrix with backtracking information. Thus, we can control and find new DTW paths in these manners:

(A2) This approach accounts for lexical variations at the end of the query. We consider cuts of up to 250ms at the end, always keeping the matching segment above 500ms (example on Figure 2). Normalized distance is obtained for all possible ending paths, and the minimum output.

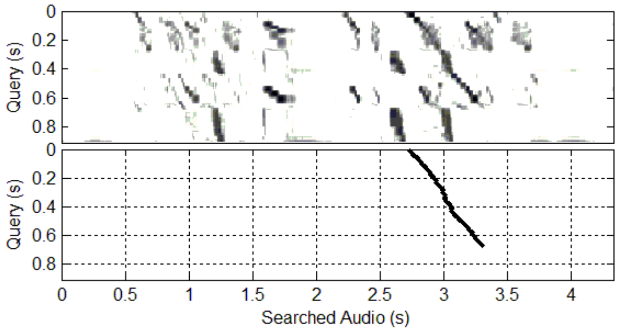

Fig 2. Query vs. Audio posterior distance matrix (top) and the best path from A2 (bottom).

(A3) This is the inverse approach of A2, by considering small lexical variations at the beginning of a query, cutting the beginning up to 250ms but keeping it above 500ms. The cumulative distance matrix doesn't directly tell the values of new possible paths, but we do not repeat the DTW. To improve computational speed, we reason that the full paths that contain the match will already be the ones with the lowest distances for this query-audio pair. Therefore, we backtrack only the 5 best paths and consider the new possible starts to get the best normalized distance possible.

(A4) This approach accounts for small extra words or filler content in the audio between the query's own words by allowing one "jump" in the DTW path (Figure 3). A jump of up to half of the query's length is allowed, and it may not occur at the initial and final 250ms and for queries shorter than 800ms.

(A5) The last approach accounts for re-ordering of query words. It allows swaps of two segments (Figure 4), and it's a similar case to A4 by allowing filler content, but the first query segment should be found ahead of the second. Since the DTW is performed left to right, we resort to backtracking the best paths as in A3, and finding an alternative path ahead of the breaking point that better matches the start of the query. An additional check is performed to ensure that no overlap occurs. The same constraints for the position and length of the jump in A4 are considered for A5's breaking point and filler content amount.

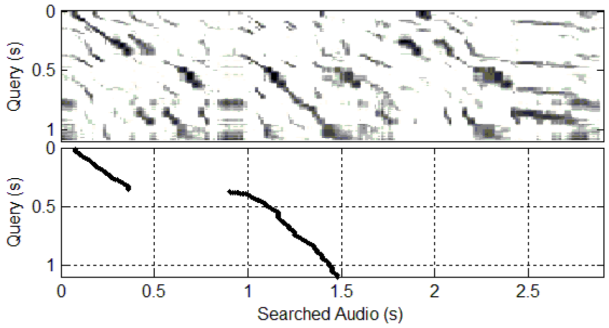

Fig. 3. Query vs. Audio posterior distance matrix (top) and the best path from A4 (bottom).

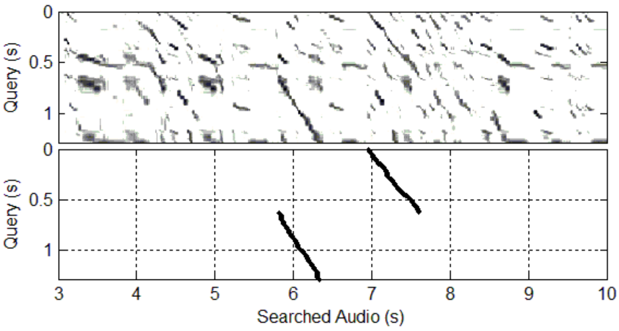

Fig. 4. Query vs. Audio posterior distance matrix (top) and the best path from A5 (bottom).

All examples shown are cases from the development dataset that were at first rejected by A1 but are now accepted with the respective strategies. The selected limitations of cuts, jumps and query durations may seem arbitrary but were selected empirically by trialing and according to some of the database specifications (such as words being larger than 5 phonemes ≈250ms).

3.4. Fusion and Calibration

From the five approaches that output distances for the same query-audio pair, it could be argued that the minimum distance obtained would correspond be the best match. However, tests showed that taking the minimum is not the best method, supposedly due to the special approaches often finding false matches. We found that the harmonic mean of the output of the approaches was a more suitable measure (best Cnxe scores on evaluation dataset: minimum 0.5256, arithmetic mean 0.5199, harmonic mean 0.5153), and is employed here for fusion systems to extract a single distance value.

A further normalization is performed per-query, by subtracting the mean and dividing by the standard deviation of all the results from Query-Audio pairs for a given

System	Dev - Cnxe, minCnxe	Eval - Cnxe, minCnxe	Dev - ATWV, maxTWV	Eval - ATWV, maxTWV
A1	0.5771, 0.5645	0.5362, 0.5252	0.4343, 0.4343	**0.4269**, 0.4291
A2	**0.5700**, 0.5568	**0.5250**, 0.5136	**0.4400**, 0.4400	0.4248, 0.4288
A3	0.5918, 0.5787	0.5531, 0.5419	0.4168, 0.4168	0.4052, 0.4082
A4	0.5883, 0.5745	0.5518, 0.5393	0.4134, 0.4134	0.4065, 0.4122
A5	0.6004, 0.5846	0.5548, 0.5411	0.4201, 0.4201	0.4183, 0.4212
Fusion 1-2	0.5637, 0.5500	0.5186, 0.5069	0.4501, 0.4501	0.4400, 0.4416
Fusion All	**0.5615**, 0.5467	**0.5153**, 0.5030	**0.4608**, 0.4608	**0.4538**, 0.4568

Table 2. Summarization of the obtained results on development (Dev) and evaluation (Eval) datasets for the five individual DTW approaches and the two fusion systems (lower Cnxe is better, higher ATWV is better).

query (standard score: $(X-\mu)/\sigma$). The results may be skewed with this step to indicate that every query should be found at least once in the data, but it is a highly beneficial procedure. Normalized distances are transformed into figures of merit by simply taking the symmetrical value.

The final step is to fuse results based on recognizers of different languages. Although there are several methods and advances in fusing classifier systems [11], we decided to employ the arithmetic mean of the already normalized values, found to be a good method on the development set. Figure 5 shows the obtained Detection Error Tradeoff (DET) curves of the best system on the development dataset for using the recognizers of three languages individually and for their fusion.

As the main evaluation metric of QUESST 2014, Cnxe is calibrated for by employing an affine transformation to the data. The linear transformation parameters are trained with the Bosaris toolkit [18] for the development set, taking into account the ground truth and the prior suggested by the task, and the linear transform is applied to the dev and eval sets. For comparison purposes, the presented minimum Cnxe (minCnxe) is computed with a stricter approach, the Pool-Adjacent Violators (PAV) transformation [19], which is non-parametric and leads to lower values of Cnxe than the affine transformation.

To get a decision if a query is a match to the audio or not, a threshold is computed by finding the maximum TWV on the dev set, using the defined miss and false alarm costs and target prior. Actual TWV (ATWV) is therefore equal to maximum TWV (maxTWV) on the development set.

4. RESULTS

We decided to analyze two fusions of DTW approaches. “Fusion All” combines the normalized distances of all five methods. “Fusion 1-2” combines only the two best individual approaches, A1 and A2. Overall results for the different systems on development and evaluation datasets of QUESST 2014 are summarized in Table 1. It is noted that, of the individual approach systems, the ones using A1 and A2 are always better performing, on dev and eval sets and for the two metrics. Even so, although close, their fusion was not better than the fusion of all approaches, which provided the best results overall for both sets and metrics. Comparing A2 and A3 (cutting the end and cutting the beginning of the query), we cannot ascertain clearly why A2 performs better, often even better than the exact matching A1. It could be due to lexical variations at the end of the query being more common (unknown in the database), or even that there are often prosodic or enunciation variations at the end of words.

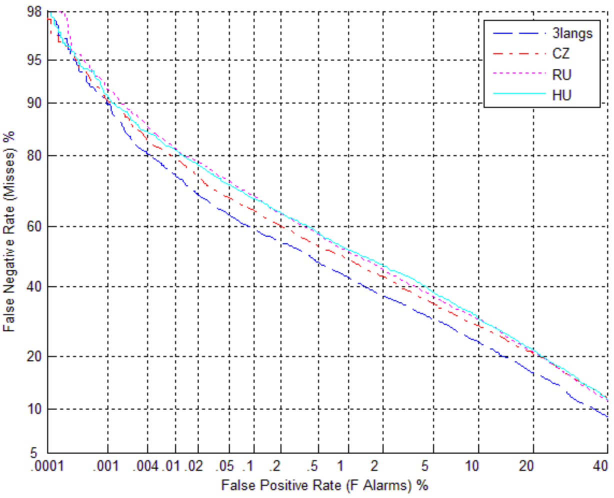

Fig. 5. Detection Error Tradeoff (DET) curves for the development dataset of systems using phonetic recognizers of Czech (CZ), Russian (RU) and Hungarian (HU) and the mean combination of results (3langs). Fusion of all 5 approaches is used.

	All types	T1	T2	T3
Fusion 1-2, Dev	0.5637	0.4124	0.6328	0.7260
Fusion All, Dev	0.5615	0.4119	0.6437	0.6876
Fusion 1-2, Eval	0.5186	**0.3959**	**0.5114**	0.7637
Fusion All, Eval	0.5153	0.3990	0.5277	**0.7089**

Table 1. Cnxe scores for the two fusion systems for the separate types of queries defined.

To further analyze the developed systems, results for the separate query types are compiled in Table 2. Although fusion 1-2 is slightly better for type 1 and type 2 queries, the fusion of all approaches was clearly helpful for the type 3 problems. These were the cases that approaches A4 and A5 targeted and, although they were indeed helpful, the complex type 3 queries were still the hardest to match. A lot more effort can be done on the detection of these matches, as we might not even have considered every possible case, such as reordering of more than 2 words.

5. CONCLUSIONS

We presented an approach to the Query-by-Example challenge of matching audio queries to multi-lingual audio documents and with possible complex query types. The DTW modifications constructed tackle the complex queries and the overall results were improved (albeit slightly) on the Mediaeval QUESST 2014 task.

For the next edition of MediaEval, it is already known that further intricate query cases will be considered for QUESST, namely spontaneously spoken queries, further approaching a real case scenario. We intend to include an additional number of phonetic recognizers for fusion as well as implementing improved fusion methods. Also, we can improve on the imposed limits for DTW path cutting or jumping (decided empirically) and we didn't consider reordering and lexical variation simultaneously, which was possible. We also intend to extend the method to the separate task of reading errors detection such as repetitions and partial words.

ACKNOWLEDGEMENTS

This work was supported in part by Fundação para a Ciênica e Tecnologia under the projects UID/EEA/50008/2013 (plurianual founding), and CATARATA PTDC/DTP-PIC/0419/2012. Jorge Proença is supported by the SFRH/BD/97204/2013 FCT Grant.

REFERENCES

[1] J.G. Fiscus, J. Ajot, J.S. Garofolo, and G. Doddington, "Results of the 2006 Spoken Term Detection Evaluation," in *Proc. SIGIR 2007 Workshop on Searching Spontaneous Conversational Speech*, Amsterdam, 2007, pp. 51–57.

[2] NIST, "OpenKWS13 Keyword Search Evaluation Plan," March 8, 2013. http://www.nist.gov/itl/iad/mig/upload/OpenKWS13-EvalPlan.pdf

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

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

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

[6] F. Metze, X. Anguera, E. Barnard, M. Davel, and G. Gravier, "Language Independent Search in MediaEval's Spoken Web Search Task," *Computer Speech and Language, Special Issue on Information Extraction & Retrieval*, vol. 28, no. 5, pp. 1066–1082, September 2014.

[7] I. Szoke, P. Schwarz, L. Burget, M. Fapso, M. Karafiat, J. Cernocky, and P. Matejka, "Comparison of keyword spotting approaches for informal continuous speech," in *Proc. Interspeech*, Lisbon, Portugal, 2005, pp. 633–636.

[8] C. Gracia, X. Anguera, and X. Binefa, "Combining temporal and spectral information for Query-by-Example Spoken Term Detection," in *Proc. European Signal Processing Conference (EUSIPCO)*, Lisbon, Portugal, 2014, pp. 1487-1491.

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

[10] L.J. Rodriguez-Fuentes, A. Varona, M. Penagarikano, G. Bordel, and M. Diez, "High-performance query-by-example spoken term detection on the SWS 2013 evaluation," in *Proc IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Florence, Italy, 2014, pp. 7819-7823.

[11] A. Abad, L.J. Rodriguez Fuentes, M. Penagarikano, A. Varona, M. Diez, and G. Bordel, "On the calibration and fusion of heterogeneous spoken term detection systems," in *Proc. Interspeech 2013*, Lyon, France, 2013, pp. 20-24.

[12] X. Anguera, L.J. Rodriguez-Fuentes, I. Szöke, A. Buzo, and F. Metze, "Query by Example Search on Speech at Mediaeval 2014," in *Working Notes Proceedings of the Mediaeval 2014 Workshop*, Barcelona, Spain, 2014.

[13] X. Anguera, L.J. Rodriguez-Fuentes, A.Buzo, F. Metze, I Szoke, and M. Penagarikano, "QUESST2014: Evaluating Query-By-Example Speech Search in a Zero-Resource Setting with Real-Life Queries," in *Proc IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Brisbane, Australia, 2015.

[14] L.J. Rodriguez-Fuentes, and M. Penagarikano, "MediaEval 2013 SpokenWeb Search Task: System Performance Measures," Technical Report-2013-1, Dept. Electricity and Electronics, University of the Basque Country, May, 2013.

[15] A. Veiga, C. Lopes, L. Sá, and F. Perdigão, "Acoustic Similarity Scores for Keyword Spotting," in *Proc. PROPOR 2014*, São Carlos, Brazil, 2014, pp. 48-58.

[16] Phoneme recognizer based on long temporal context, Brno University of Technology, FIT. http://speech.fit.vutbr.cz/software/phoneme-recognizer-based-long-temporal-context

[17] P. Pollak, et al., "SpeechDat(E) – Eastern European Telephone Speech Databases," in *Proc. of XLDB 2000, Workshop on Very Large Telephone Speech Databases*, Athens, Greece, 2000.

[18] N. Brummer, and E. de Villiers, "The BOSARIS Toolkit User Guide: Theory, Algorithms and Code for Binary Classifer Score Processing," Technical report, 2011. https://sites.google.com/site/bosaristoolkit/

[19] N. Brummer and J. du Preez, "The PAV algorithm optimizes binary proper scoring rules," arXiv:1304.2331, 2013.

[20] The 2014 Query by Example Search on Speech (QUESST) http://www.multimediaeval.org/mediaeval2014/quesst2014/