Ingénierie des Systèmes d'Information
Vol. 29, No. 3, June, 2024, pp. 1175-1181
Journal homepage: http://iieta.org/journals/isi

Self-Supervised Spoken Term Detection for Query by Example

Nadia Benati[1,2,3*], Halima Bahi[1,2]

[1] LISCO Laboratory Badji Mokhtar-Annaba University, Annaba 23000, Algeria
[2] Computer Science Department, Badji Mokhtar-Annaba University, Annaba 23000, Algeria
[3] Computer Science Department, Mohamed-Cherif Messaadia University, Souk Ahras 41000, Algeria

Corresponding Author Email: nadia.benati@univ-annaba.dz

https://doi.org/10.18280/isi.290334

Received: 23 October 2023
Revised: 20 May 2024
Accepted: 2 June 2024
Available online: 20 June 2024

Keywords:
attention mechanism, Fully Connected Convolutional Neural Network (FC-CNN), query by example, self-supervised learning, speech search, spoken term detection

ABSTRACT

Technological advances in signal processing and enormous data storage capabilities have made large speech datasets readily available. These speech data often being unlabelled and unannotated their effective use is a research hot topic. In particular, query by example spoken term detection (QbE-STD) constitutes a potential way to take advantage of these resources. Given a speech stream, QbE-STD techniques aim to detect the presence of a given spoken query, in an unsupervised way. QbE-STD is considered a pattern-matching problem, and the dynamic time-warping (DTW) algorithm is state-of-the-art. This paper proposes an alternative to the DTW and its variants by leveraging deep learning models. The proposed architecture uses convolutions to extract short-term dependencies and an attention mechanism to extract long-term dependencies, from the speech archive. Thus, a Fully Connected Convolutional Neural Network (FC-CNN) is trained to detect the presence of a target spoken query in a speech stream. During the training stage, the model is trained on raw speech, in a self-supervised way, while during the test stage and without prior knowledge on the target query, the system is expected to decide whether the query exists or not within the speech archive. Experiments were carried on a dataset built on the Google speech commands corpus. The obtained results show the potential of our suggestion in indexing speech databases.

1. INTRODUCTION

The era of big data overwhelmed us with data from various sources and in several formats. Among the easiest data to capture but the least used is speech. Herein, the spoken term search task paved the way for several applications to leverage this huge amount of resources.

Spoken term detection (STD) refers to automatically searching for some interesting speech segments in a continuous audio stream of speech [1, 2]. This technology is increasingly used in many fields, including indexing and searching in multimedia archives, video lectures, automated call centers, voice-based human-computer interfaces, voice recognition, etc. When it comes to dealing with specific words, STD refers to keyword spotting (KWS) [3, 4] otherwise it is query by example spoken term detection (QbE-STD) [5, 6].

An intuitive way to implement QbE-STD systems is to develop very large vocabulary speech recognizers. Meanwhile, the development of robust recognizers requires the availability of a huge amount of annotated and labeled speech data. Even these systems can be inefficient as in the case of spontaneous speech, or a multilingual context [1]. Moreover, in the case of indexing and searching in multimedia archives, the spoken query is not previously known, thus, QbE-STD is addressed as an unsupervised task [5]. In this work, we assume that no prior knowledge is available on the spoken term. In such cases, QbE-STD is considered a pattern-matching problem, and it involves two main stages: feature extraction, and pattern matching.

In the feature extraction stage, features are extracted from the spoken query as well as from the test utterance. For that, the Mel frequency cepstral coefficients as well as posteriorgrams have been widely used [2-7].

Recent advances in deep learning models favored deep neural networks like autoencoders (AE) and convolutional neural networks (CNN) in extracting embedding features from the raw speech [8, 9].

The pattern-matching stage aims to assess the similarity between the spoken query and the speech utterance to investigate. Here, the dynamic time warping (DTW) algorithm is state-of-the-art [1, 6, 7,10, 11]. DTW as well as its variants were widely criticized due to their slowness [11], and due to the high rate of false alarms they involve. Meanwhile, one of the important issues in query-by-example applications is to find an adequate representation of the spoken query [11].

In this work, we avoid similarity distance computation and take advantage of deep neural network models to represent the speech and to detect the presence of the spoken query. At the same time, we consider real conditions where the term to search is not previously known, and there is only a single occurrence of it, that produced during the query.

We suggest handling the QbE-STD problem as a

classification problem using a Fully Connected Convolutional Neural Network (FC-CNN) model trained in a self-supervised way to detect the presence/absence of the incoming spoken query within the test utterance from speech archives.

As CNN models are inherently prone to be trained in a supervised way, an artificial training dataset is built where a spoken query is added at the end of the speech archive to prospect. An attention mechanism that focuses on this part of the pattern is added. When the spoken term exists in the test utterance, the built sample has at least two utterances of the spoken query, otherwise, the spoken query exists once in the produced speech file. During the test stage, the model is expected to detect the single or multiple occurrences of the spoken query, standing respectively for the presence or absence of the spoken query.

This paper is organized as follows. The next section outlines the previous approaches for query by example spoken term detection. The description of the proposed detection method is detailed in Section 3. Section 4 describes the constructed dataset and introduces the experimental results. At the end, conclusions and future works are drawn.

2. LITERATURE REVIEW

Before the feature extraction stage, pre-processing steps take place. In this case, the pre-processing phase includes signal normalization and silence removal from the input speech.

Normalization aims to reduce the variability within the speech signal and makes the data more consistent for the training of the deep learning models [12]. In particular, normalization is used to avoid overfitting, a problem hardly faced during the training stage.

Meanwhile, silence removal is usually used in speech processing applications when using machine learning approaches as the silence carries non-informative information and may slow the training process.

2.1 Feature representation

For spoken term detection purposes, features are represented in different forms. These representations are most of the time related to the primary features like Mel frequency cepstral coefficients (MFCC) or perceptual linear prediction (PLP) features.

Besides that, posteriorgrams have been widely used in QbE-STD applications. Given an acoustic vector, the posteriorgram represents its posterior distribution over a set of predefined classes. The main applied posteriorgram representations, in this context, are Gaussian posteriorgrams [13] and phonetic posteriorgrams [11]. A Phonetic posteriorgram is a time/class matrix, where each cell value is the posterior probability of a phone class for the associated time frame. Gaussian posteriorgram is a vector of probabilities representing the posterior probabilities of a collection of Gaussian components for a speech window. Gaussian mixture models (GMMs) are trained on raw speech or on primary features in an unsupervised way to label the speech frames [11, 14, 15].

With the rise of deep learning models, acoustic word embeddings (AWEs), as well as bottleneck and articulatory bottleneck features, have been applied as an alternative to Gaussian posteriorgrams [15-17]. Meanwhile, architectures such as convolutional neural networks have proven their effectiveness in extracting local properties in speech signals [18].

2.2 Pattern matching algorithms

Broadly, the pattern-matching stage computes the similarity between the feature representation of the spoken query and that of the test utterance [19]. Herein, the widely used method for template matching is the DTW algorithm [20].

DTW is utilized in the comparison between two vectors that mainly do not have the same dimensionality. In our case, DTW pertains to acoustic vectors representing speech segments. Particularly, it emphasizes regions in the speech archive that resemble the target query. Herein, template matching is applied either by using spectral-based features such as MFCCs and PLPs or by other means such as the aforementioned posterior characteristics.

To make it more suitable for audio search applications, several modifications have been made to the basic DTW algorithm. Segmental DTW (S-DTW) was proposed to realize feature sequence matching between the target query and the speech archive [13]. In S-DTW, the sequence corresponding to the archive to investigate is divided into overlapped vectors that have the same length as the target query. Therefore, the algorithm requires to be executed over all the vectors to detect the target query pattern, which increases its computational needs. To reduce the time complexity, other variants were proposed such as subsequence DTW (sub-DTW) [21] or non-segmental [22]. Distance-based approaches require a threshold determination, which was addressed by using many tuning methods.

In recent works, a similarity matrix is computed using the cosine distance between the speech archive representation and the query representation vectors, herein, the matching regions are detected using image processing methods, leading to the emergence of a diagonal pattern within the similarity matrix when transformed into an image [9, 23], as illustrated in Figure 1. It is the so-called diagonal pattern search method [24].

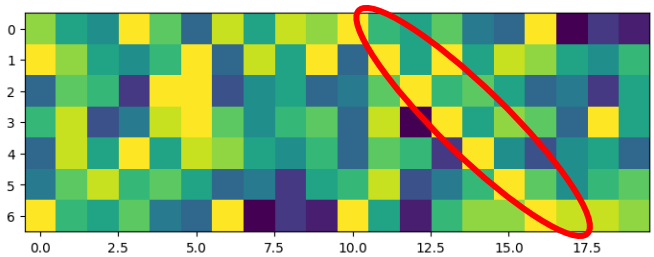


Figure 1. The diagonal in the similarity matrix image stands for the spoken term detection

2.3 Related work

With the rise of deep learning (DL), many models were applied for KWS purposes [25], however, only a few works dealt with QbE-STD.

Chen et al. [15] proposed to train a deep neural network (DNN) to find useful characteristics in unlabeled speech data. In this work, labels are generated in an unsupervised way, by a Dirichlet GMM model. The DNN trained with these labels is then used to extract low-dimensional features, called unsupervised bottleneck features (uBNFs), which serve as input for the clustering stage where different groups of sounds are determined. The acoustic vectors from the archive and the

query are compared using the subDTW algorithm.

To address the background noise issue, Lim et al. [16] suggested the use of a CNN to extract the bottleneck features.

To cope with low-resource scenarios and the substantial computation involved by AWEs, in the study by Yuan et al. [26], suggested the use of binary embeddings with hamming distance; their suggestion accelerates the search by leveraging efficient bitwise operations.

To address the challenge related to speech variability, Sudhakar et al. [27] suggested the RASTA-PLP spectrogram as the acoustic feature representation, then the aforementioned diagonal pattern search method computed the similarity between the target term and the speech archive. The tests they carried on the IITKGP-SDUC speech corpus achieved accuracies of about 60.2% and 70.6% for Hindi and Bengali languages respectively.

Following an end-to-end approach, Ao and Lee [28] represented the audio segments as MFCC coefficients, and the obtained vectors are sequentially processed by a long short-term memory (LSTM) model combined with an attention mechanism. The proposed network produces a scalar standing for the probability that the spoken query belongs to the test utterance.

Broadly, in the era of DL, acoustic word embeddings have been established as the preferred fixed-length vector representation for speech utterances, and the diagonal pattern search method based on CNN tends to replace the standard pattern matching methods [24].

3. PROPOSED APPROACH

The proposed solution represents an alternative to the previous approaches, and it fits low-resource scenarios as it does not need prior knowledge on the target language and avoids pattern-matching computations. The proposition leverages advances made in computer vision as the speech signals are raw speech and the detection of the query spoken term is addressed as a binary image classification problem.

Given a speech archive to prospect, the speech stream is split into segments of about a few seconds, then each segment is investigated to detect the presence (or not) of the spoken incoming query. When the spoken term to detect is provided to the detection system, the speech archive and the target term are concatenated and fed to the detection system and processed as an image by an end-to-end convolutional neural network.

As the CNN model is inherently a DNN model trained in a supervised way, we suggest training it in a self-supervised manner by creating an artificial dataset. The training dataset is divided into two parts to make it balanced among positive and negative classes.

The first part includes speech segments where the spoken query exists within the speech archive. Such samples are built by adding a target spoken query at the end of the speech archive that already includes a realization of the target term, thus, the new speech segment includes two or more realizations of the target spoken term.

Figure 3 illustrates throughout the spectrogram when an archive that contains a spoken query is concatenated to another realization of this query. In this example, the test utterance is illustrated by the file “narrative1.wav” from the IPA archive accessible at (https://www.internationalphoneticassociation.org/content/ipa-handbook-downloads), its content corresponds to the text: “The north wind and the sun were disputing which was the stronger, when a traveler came along wrapped in a warm cloak”. While the spoken query corresponding to the word “traveler” is extracted from the “narrative5.wav” recording. When the two signals are concatenated, it can be seen that the resulting spectrogram has two realizations of the same object.

The QbE-STD proposed system is built upon the CNN model as a feature extractor and the fully connected layers as a classifier, trained in a self-supervised way. Figure 2 provides an overview of the proposed QbE-STD system.

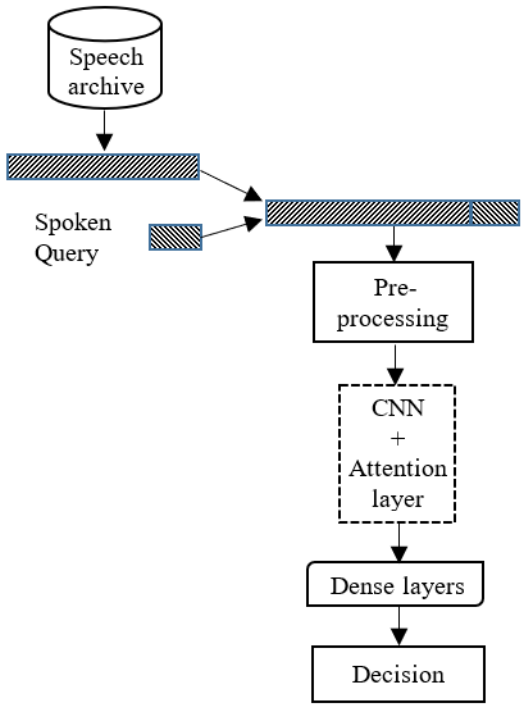


Figure 2. Overview of the QbE-STD proposed system

The second part of the dataset contains speech segments illustrating the case where the target spoken query does not exist within the speech archive. Herein, the target spoken query is added at the end of the speech archive that does not contain any realization of this term, leading to segments that have only one occurrence of the query.

The detection system is trained on the built dataset. During the test stage, considering the archive to investigate and an unknown incoming query, the query is concatenated to the archive and fed to the trained model. The system is expected to detect if the last item already exists in the archive or not.

Therefore, the detection stage is handled as a binary classification problem and is implemented through the fully connected layers.

3.1 Feature extraction and representation

The presence of spectral variations and local correlations in speech signals makes CNNs the most appropriate DNN model to address speech processing applications. The raw speech is fed to the CNN model to extract pertinent characteristics. The CNN is composed of four consecutive blocks, each of which includes: a convolutional layer, a max pooling layer, and a dropout layer. The proposed sequence of convolutional blocks is shown in Table 1. This architecture is inspired by that of AlexNet [29] which met great success in speech recognition applications [30].

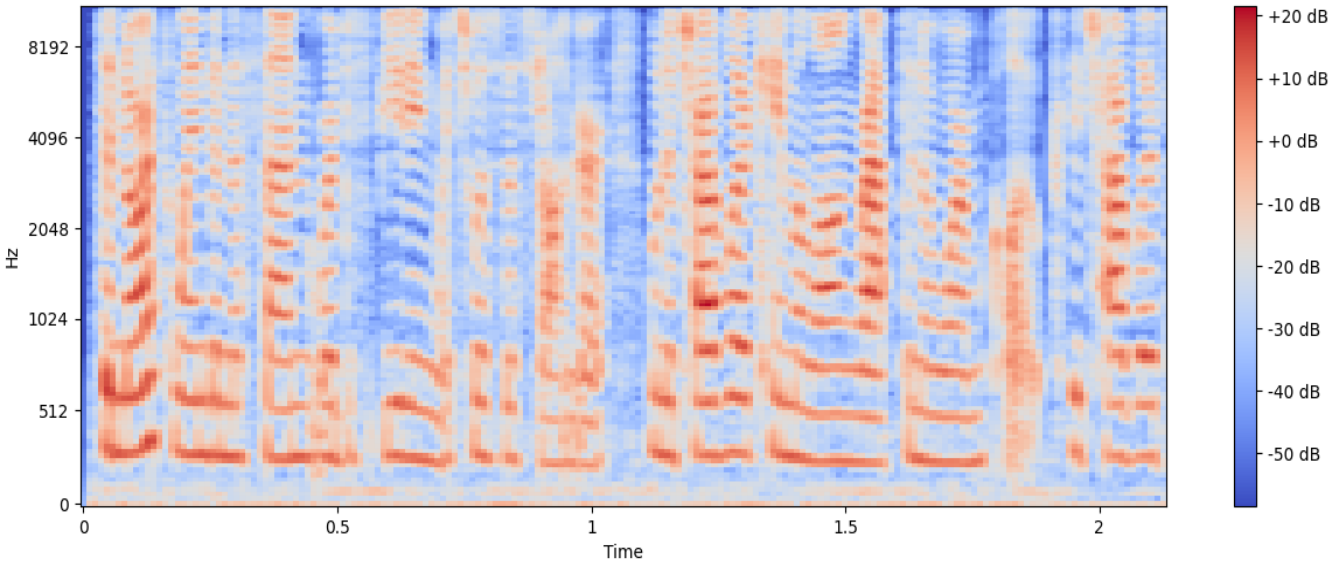


Figure 3. Spectrogram of the "narrative1.wav" recording concatenated with the term "traveler"

Table 1. The convolutional blocks architecture

#First Conv1D Layer	#Second Conv1D Layer	#Third Conv1D Layer	#Fourth Conv1D Layer
Conv1D (8, 13)	Conv1D (16, 11)	Conv1D (32, 9)	Conv1D (64,7)
MaxPooling1D	MaxPooling1D	MaxPooling1D	MaxPooling1D
Dropout	Dropout	Dropout	Dropout

Indeed, the number of filters increases in the deeper layers of the network, while the size of the windows decreases when the network goes deeper. In fact, the first layers capture low-level features such as simple patterns whereas deeper ones are intended to capture more complex patterns.

3.2 Attention mechanism

The attention mechanism allows DL models to focus on specific parts of the input, improving their ability to understand complex patterns. Attention mechanisms have been specifically used in natural language processing (NLP) contexts, such as text understanding or translation [31], where the attention mechanism has shown great success in implementing selective memory.

Meanwhile, CNNs are known to be powerful feature extractors, however, they treat all the image parts equally.

In our suggestion, the spoken term is artificially added at the end of the image, thus, it would be valuable if particular attention is given to this part. This is the vocation of the attention mechanism. It is used in conjunction with the CNN layers, and its output is used as the input of the dense layers.

3.3 Classification

Once the features are extracted from the raw speech, the obtained acoustic vector is fed to the subsequent dense layers. The last layer has one neuron standing for the two alternatives, namely: the spoken term detected, and not detected. The FC-CNN model includes three consecutive Dense, as shown in Table 2.

The function "tanh" was used as the activation function for all layers, except for the output layer where the "sigmoid" was used. The learning rate was 0.0005 with Adam optimizer; and to avoid overfitting, L1-regularization of 0.001 was used at the several model layers.

Table 2. The fully connected blocks architecture

#First Dense Layer	#Second Dense Layer	#Third Dense Layer
Dense (256)	Dense (128)	Dense (1)
Dropout (0.2)	Dropout (0.2)	

4. EXPERIMENTAL RESULTS

4.1 Dataset

A dedicated dataset was artificially created upon the well-known Google speech commands, freely accessible at http://download.tensorflow.org/data/speech_commands_v0.02.tar.gz.

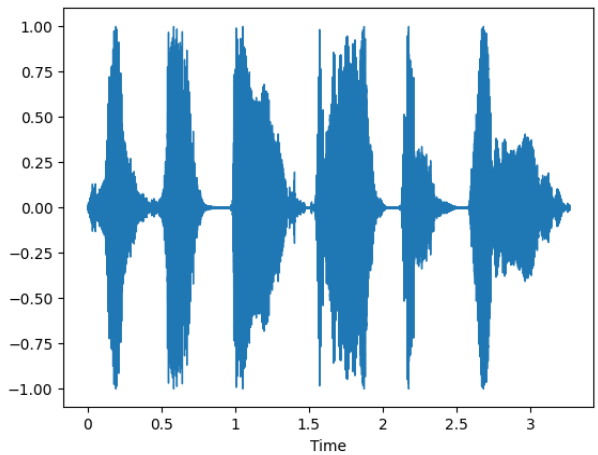


Figure 4. Example of an archive segment with the words: "Right-cat-bird-three-down-visual"

The Google speech commands corpus, designed to help

implement keyword-spotting systems, is composed of over 100,000 one-second-long utterances of 30 words uttered by thousands of different speakers.

For this study, a collection of words from the dataset were concatenated to form the test utterance speech stream, as illustrated in Figure 4. The balanced training set includes 2000 samples, each of them of about 5-8s.

While, in QbE-STD systems, the choice of the spoken query is a challenging task [5], in our case, the spoken query was randomly chosen from the 30 available words. This brings the detection closer to real-life conditions but makes it more difficult. The chosen spoken query is then added to the archive signal to form the wave file to include in the new dataset.

4.2 Results

The target applications, such as speech archive indexing or multimedia search, are prone to favor accurate detection and avoid false alarms. Thus, the evaluation process is mainly based on accuracy, false alarm rate, precision, and recall scores, given that:

True positive (TP) is related to the number of rightly detected queries.

False positive (FP) is related to the number of falsely detected queries

True negative is related to the number of rightly undetected queries.

False negative is related to the number of undetected queries.

Accuracy measures how often the classifier detects rightly the presence or the absence of the spoken query, it is defined as:

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

The false alarm rate is defined as the ratio between the falsely detected queries and the whole samples where the query is absent.

$$FAR = \frac{FP}{TN + FP} \quad (2)$$

Precision is the ratio of rightly detected queries to the total number of hits (detection). It is defined as:

$$precision = \frac{TP}{TP + FP} \quad (3)$$

Recall stands for the true positive rate, defined as:

$$recall = \frac{TP}{TP + FN} \quad (4)$$

Finally, the F1 measure represents the harmony between the precision and the recall, it is computed as:

$$F1 - score = \frac{2 \times precision \times recall}{precision + recall} \quad (5)$$

Table 3 reports the results obtained over the 2000 samples where the training dataset represents 70%, the validation dataset 10%, and the test dataset 20% of the built corpus.

The obtained results show an accuracy of about 72.5% on the test dataset, which is promising for indexing applications, with a false alarm rate of about 25%.

Additional experiments were carried on the test dataset to compare the proposed method to the standard DTW algorithm, results show an accuracy of about 69% for the latter.

Table 3. Obtained results

	Training	Test
Accuracy (%)	87	72.5
False alarm rate (%)	12.6	25
Precision (%)	87.4	73
Recall (%)	87.75	70
F1-score (%)	87.57	71.47

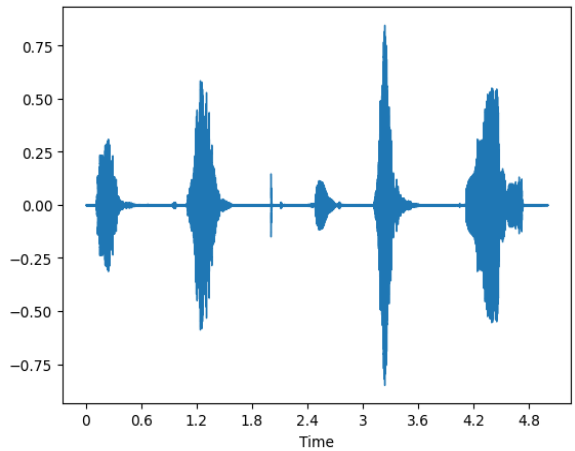
4.3 Discussion

The proposed approach is not a distance-based one and is completely carried out in an unsupervised mode. Indeed, the spoken query is not known in advance, moreover, it is chosen randomly as well as the related speaker.

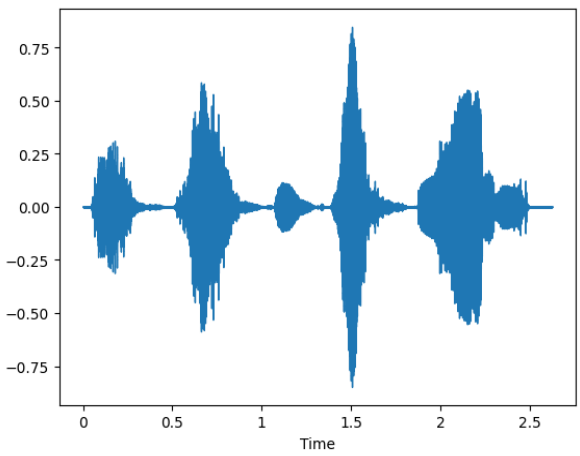
The experiments show the potential of the proposed approach. The low accuracy can be attributed to the high variability between the speakers seen during the training stage and those of the test stage.

The pre-processing stage which removes silence and thus fastens the training stage, also slightly affects the model performances. Figure 5 illustrates this effect.

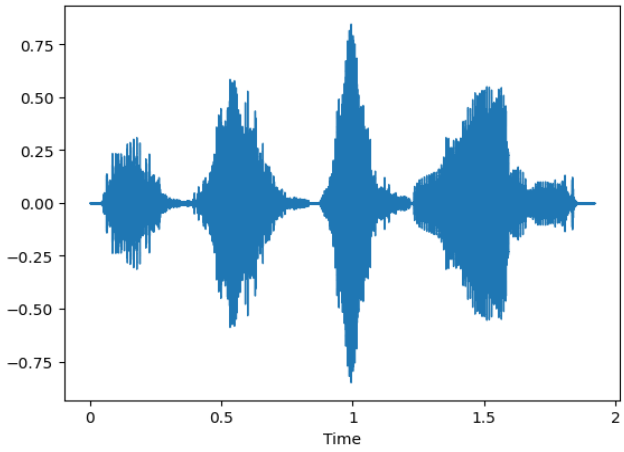
Figure 5(a) shows the original signal of the artificially built sequence “off-right-house-yes-learn”. Figure 5-b illustrates the case where the silence is removed at a threshold of 20 dB. In Figure 5-c the sound loudness is lessened to 18 dB, leading to the deletion of the word ‘house’.



(a)



(b)



(c)

Figure 5. Illustration of silence removal (a) the original signal (b) silence removed with top = 20dB (c) silence removed with top = 18dB

Therefore, when the top for speech removal is inadequate, words are removed which may represent the spoken query to detect.

Meanwhile, the random choice of the spoken query makes our system suitable for real-life applications but heavily contributes to reducing its accuracy.

5. CONCLUSIONS

Speech archives have significantly grown in recent years, making audio database indexing and searching promising applications. For that purpose, unsupervised spoken term detection for query by example is of great importance. In this study, a self-supervised QbE-STD system, that aims to overcome limitations related to state-of-the-art pattern-matching approaches, is presented.

Our suggestion is independent of language and speakers, which makes it suitable for low-resource languages. To evaluate the proposed method, extensive experiments are conducted. The obtained results show the potential of the proposed approach in the context of a randomly chosen spoken query, and in a variable context where several speakers are involved in the speech stream and where the query is uttered by an external speaker.

While the obtained results are promising, much more tuning has to be investigated to improve the outcomes.

Finally, the present work showed an interest in deep learning approaches for QbE-STD both for feature extraction and detection stages. Therefore, deep neural networks trained in a supervised or unsupervised way need to be deeply investigated for this purpose.

REFERENCES

[1] Deekshitha, G., Mary, L. (2020). Multilingual spoken term detection: A review. International Journal of Speech Technology, 23(3): 653-667. https://doi.org/10.1007/s10772-020-09732-9

[2] Mizuochi, S., Nose, T., Ito, A. (2022). Spoken term detection of Zero-Resource language using posteriorgram of multiple languages. Interdisciplinary Information Sciences, 28(1): 1-13. https://doi.org/10.4036/iis.2022.A.04

[3] Bahi, H., Benati, N. (2009). A new keyword spotting approach. In 2009 International Conference on Multimedia Computing and Systems, Ouarzazate, Morocco, pp. 77-80. https://doi.org/10.1109/MMCS.2009.5256728

[4] Benayed, Y., Fohr, D., Haton, J.P., Chollet, G. (2002). Keyword spotting using support vector machines. In Fifth International Conference on Text, Speech and Dialogue-TSD 2002, Brno, Czech Republic. https://doi.org/10.1007/3-540-46154-X_39

[5] Mandal, A., Prasanna Kumar, K.R., Mitra, P. (2014). Recent developments in spoken term detection: A survey. International Journal of Speech Technology, 17: 183-198. https://doi.org/10.1007/s10772-013-9217-1

[6] Helén, M., Lahti, T. (2006). Query by example methods for audio signals. In Proceedings of the 7th Nordic Signal Processing Symposium-NORSIG 2006, Reykjavik, Iceland, pp. 302-305. https://doi.org/10.1109/NORSIG.2006.275240

[7] Wang, H., Lee, T., Leung, C.C. (2011). Unsupervised spoken term detection with acoustic segment model. In 2011 International Conference on Speech Database and Assessments (Oriental COCOSDA), Hsinchu, Taiwan, pp. 106-111. https://doi.org/10.1109/ICSDA.2011.6085989

[8] Mantena, G., Prahallad, K. (2014). Use of articulatory bottle-neck features for query-by-example spoken term detection in low resource scenarios. In 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Florence, Italy, pp. 7128-7132. https://doi.org/10.1109/ICASSP.2014.6854983

[9] Ram, D., Miculicich, L., Bourlard, H. (2018). CNN based query by example spoken term detection. In Interspeech, pp. 92-96. https://doi.org/10.21437/Interspeech.2018-1722

[10] Benati, N., Bahi, H. (2016). Spoken term detection based on acoustic speech segmentation. In 2016 7th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT), Hammamet, Tunisia, pp. 267-271. https://doi.org/10.1109/SETIT.2016.7939878

[11] Madhavi, M.C., Patil, H.A. (2018). Design of mixture of GMMs for query-by-example spoken term detection. Computer Speech & Language, 52: 41-55. https://doi.org/10.1016/j.csl.2018.04.006

[12] Sefara, T.J. (2019). The effects of normalisation methods on speech emotion recognition. In 2019 International Multidisciplinary Information Technology and Engineering Conference (IMITEC), Vanderbijlpark, South Africa, pp. 1-8. https://doi.org/10.1109/IMITEC45504.2019.9015895

[13] Zhang, Y., Glass, J.R. (2009). Unsupervised spoken keyword spotting via segmental DTW on Gaussian posteriorgrams. In 2009 IEEE Workshop on Automatic Speech Recognition & Understanding, Moreno, Italy, pp. 398-403. https://doi.org/10.1109/ASRU.2009.5372931

[14] Chan, C.A., Lee, L.S. (2013). Model-based unsupervised spoken term detection with spoken queries. IEEE Transactions on Audio, Speech, and Language Processing, 21(7): 1330-1342. https://doi.org/10.1109/TASL.2013.2248714

[15] Chen, H., Leung, C.C., Xie, L., Ma, B., Li, H. (2016).

Unsupervised bottleneck features for low-resource query-by-example spoken term detection. In Interspeech , pp. 923-927. https://doi.org/10.21437/Interspeech.2016-313

[16] Lim, H., Kim, Y., Kim, Y., Kim, H. (2017). CNN-based bottleneck feature for noise robust query-by-example spoken term detection. In 2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), Lumpur, Malaysia, pp. 1278-1281. https://doi.org/10.1109/APSIPA.2017.8282220

[17] Yuan, Y., Leung, C.C., Xie, L., Chen, H., Ma, B., Li, H. (2017). Pairwise learning using multi-lingual bottleneck features for low-resource query-by-example spoken term detection. In 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, USA, pp. 5645-5649. https://doi.org/10.1109/ICASSP.2017.7953237

[18] Sudhakar P, Sreenivasa Rao K. Pabitra M. (2023). Query-by-example spoken term detection for zero-resource languages using heuristic search. acm trans. Asian Low-Resour. Language Information Process. https://doi-org.sndl1.arn.dz/10.1145/3609505

[19] Bridle, J.S. (1973). An efficient elastic-template method for detecting given words in running speech. In British Acoustical Society Meeting, 2: 1-4.

[20] Sakoe, H., Chiba, S. (1978). Dynamic programming algorithm optimization for spoken word recognition. IEEE Transactions on Acoustics, Speech, and Signal Processing, 26(1): 43-49. https://doi.org/10.1109/TASSP.1978.1163055

[21] Miller, D.R.H., Kleber, M., Kao, C.L., Kimball, O., Colthurst, T., Lowe, S.A., Schwartz, R.M., Gish, H. (2007). Rapid and accurate spoken term detection. In Eighth Annual Conference of the International Speech Communication Association, pp. 314-317. https://doi.org/10.21437/Interspeech.2007-174

[22] Rodriguez-Fuentes, L.J., Varona, A., Penagarikano, M., Bordel, G., Diez, M. (2014). High-performance query-by-example spoken term detection on the SWS 2013 evaluation. In 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Florence, Italy, pp. 7819-7823. https://doi.org/10.1109/ICASSP.2014.6855122

[23] Naik, P., Gaonkar, M.N., Thenkanidiyoor, V., Dileep, A.D. (2020). Kernel based matching and a novel training approach for CNN-based QbE-STD. In 2020 International Conference on Signal Processing and Communications (SPCOM), Bangalore, India, pp. 1-5. https://doi.org/10.1109/SPCOM50965.2020.9179588

[24] Kumar, K.R. Rao, K.S. (2022). A novel approach to unsupervised pattern discovery in speech using Convolutional Neural Network. Computer Speech & Language, 71: 101259. https://doi.org/10.1016/j.csl.2021.101259

[25] López-Espejo, I., Tan, Z.H., Hansen, J.H., Jensen, J. (2021). Deep spoken keyword spotting: An overview. IEEE Access, 10: 4169-4199. https://doi.org/10.1109/ACCESS.2021.3139508

[26] Yuan, Y. Xie, L., Leung, C.C. Chen, H. Ma, B. (2020) Fast query-by-example speech search using attention-based deep binary embeddings. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 28: 1988-2000. https://doi.org/10.1109/TASLP.2020.2998277

[27] Sudhakar, P., Sreenivasa Rao, K., Mitra, P. (2023). Unsupervised discovery of recurring spoken terms using diagonal patterns. In International Conference on Pattern Recognition and Machine Intelligence, Kolkata, India, pp. 61-69. https://doi.org/10.1007/978-3-031-45170-6_7

[28] Ao, C.W., Lee, H.Y. (2018). Query-by-example spoken term detection using attention-based multi-hop networks. In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, pp. 6264-6268. https://doi.org/10.1109/ICASSP.2018.8462570

[29] Krizhevsky, A., Sutskever, I., Hinton, G.E. (2012). Imagenet classification with deep convolutional neural networks. Communications of the ACM, 60(6): 84-90. https://doi.org/10.1145/3065386

[30] Talai, Z., Kherici, N., Bahi, H. (2023). Comparative study of CNN structures for Arabic speech recognition. Ingénierie des Systèmes d'Information, 28(2): 327-333. https://doi.org/10.18280/isi.280208

[31] Vaswani, A., Shazeer, N., armar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I. (2017). Attention is all you need. In 31st Conference on Neural Information Processing Systems, Long Beach, CA, USA, 30: 5998-6008. https://typeset.io/papers/attention-is-all-you-need-1hodz0wcqb.