An Unsupervised Spoken Term Detection System for Urdu

Hafiz Rizwan Iqbal, Saad Bin Zahid, Agha Ali Raza
Information Technology University, Lahore, Pakistan
{rizwan.iqbal, mscs15019, agha.ali.raza}@ itu.edu.pk

Abstract

Over the past 40 years, Keyword Spotting (KWS) remained in focus by both academia and commercial companies. However, the majority of these systems were developed and evaluated for rich resourced languages like English, German, etc. This is because it requires thousands of hours of transcribed speech data to train KWS systems, which is not available for most of the under-resourced languages like Urdu. To address this challenge, the area of zero-resource or unsupervised speech processing emerged, i.e. to extract meaningful features and learning language structures directly from unlabeled raw speech data. This paper presents a completely unsupervised KWS system that searches all of the instances of an input keyword in reference audio file(s), given the keyword present in the reference file(s), without requiring any labeled data and speech recognition. PRUS corpus was used to train GMM without any supervision. Input keyword and reference audios Gaussian Posteriorgrams were compared using Segmental Dynamic Time Warping (SDTW). Top N minimum distances were taken to obtain the closely related segments of the reference file, which are more probable to be the desired keyword. The proposed system showed the precision up to 91.50 % and 79.20 % for cross-speaker and same speaker respectively.

1. Introduction

Spoken Term Detection (STD) a.k.a. Keyword Spotting (KWS) is a task of automatically detecting a spoken term (referred to as Query) along with its location within a continuous speech. It is on the rise due to its variety of applications such as shortlisting of audios from large repositories of online lectures like Coursera [27], conference recordings (e.g. TED talks), radio and television archives. Wake-word applications (to activate or initiate a voice interaction with devices), phone call monitoring and routing are some other important applications of KWS.

STD remained a hot research topic for more than four decades, and a lot of methods have been proposed which can be categorized into 1) Large Vocabulary Continuous Speech Recognition methods (LVCSR) – used for audio indexing and speech data mining, 2) Keyword/Filler Methods a.k.a. Acoustic Keyword Spotting and 3) Query-by-Example (QbyE) method. However, the majority of these techniques were developed and evaluated for resource-rich languages like English, German, etc. because of their reliance on thousands of hours of transcribed audio data. For example, in traditional Keyword/Filler models, word/phone level transcribed data is required to train a speech recognizer [8] [12] [21].

Unfortunately, such resources are not available for many of the world's languages such as Urdu. With the recent development of the internet, media technologies and smartphones, it is quite easy to obtain audio data than the transcription work. It's not only a time taking activity; but also requires a reasonable level of linguistic knowledge for performing annotations. This is the reason that most of the academic and commercial organizations develop STD systems for a few hundred languages [26].

As we are living in a digital and communication age in which digital media can be produced and gathered at a pace that far surpasses our capacity to transcribe it, a common question "how much can be directly learned from the speech signals alone, without any supervision"? In addition to this, speech applications are becoming popular and available for many languages on the cost of increasing method complexities and their dependency on transcribed resources [23], it is difficult to envision that the required resource collections would cover all 7,000 human languages around the globe [2]. This makes a related query that "what unsupervised techniques can be performed well in contrast to the traditional supervised training techniques". This creates a related question "what techniques can be performed well using unsupervised techniques in comparison to more conventional supervised training methods". Our motivation to answer these two questions lead us to explore the development of an unsupervised STD system for a low resource language Urdu.

Urdu is the 6th most popular Asian language, the national language of Pakistan and the authorized language of 6 Indians states with more than 175 million speakers all over the world [31]. As the national language of Pakistan, most of the educational material, radio and television programs, and conversational audios are available in Urdu. This plethora of available speech files creates a need for an efficient KWS for Urdu language. Limited efforts have been made in the past [17], but unfortunately, there are no publicly available automatic KWS for Urdu.

This paper presents an unsupervised STD system for Urdu language. To train the model from an unlabeled speech data, a Gaussian Mixture Model (GMM) was used to represent each audio frame with a Gaussian Posteriorgram (GP) vector, and a Segmental Dynamic Time Warping (SDTW) method is used to compare the GPs of the spoken query term (hereinafter called *Needle*) and the target speech utterance (hereinafter called *Haystack*) [36] to find one or more occurrences of the *needle*.

In addition to this, the proposed KWS system searches all of the instances of the *needle* with their locations in the *haystack(s)* without doing speech recognition, given the keyword is present in the reference file. For this purpose, a Phonetically Rich Urdu Speech (PRUS) Corpus [37] used to cluster speech frames without any transcribed data. Top *N* minimum distances were then taken to get the closely related segments of the *haystack* file with the assumption that these speech frames are the most probable to be the desired keyword. The proposed system showed the *precision* up to *91.50%* and *79.20%* for cross and the same speaker respectively, given the *needle* is present in the *haystack*.

2. Literature Review

STD has been a hot research area over the past 4 decades but in recent years STD has received increased attention by both academia and commercial communities [38]. Chen et.al [3] summarizes STD past research efforts and encapsulates proposed methods in three categories.

The first category defines Large Vocabulary Continuous Speech Recognition (LVCSR) based methods. LVCSR based methods have been extensively used in audio data mining and indexing and found to be well accurate for a variety of tasks [39]. Continuous speech files are transcribed into words using Automatic Speech Recognizer (ASR) and then text-based searching techniques used for efficient spotting of the required keywords [40].

The second type of STD methods are Keyword or Filler methods *aka* Acoustic Keyword Spotting, models the keywords and non-keywords using Hidden Markov Models (HMMs) and spotting is made through the decoding graphs where keywords and fillers appear in parallel [42] [43] [44]. This type of KWS mostly used in scenarios where keywords are pre-defined and speech data comes in real-time. Such types of applications are like voice commands and wake word applications (e.g. Hey Siri, Ok Google, etc.). Ketabdar et.al [14] proposed a system that used the HMMs posterior based scoring approach for keyword and non-keyword elements [7]. For each frame, the state posteriors are combined with the posteriors of keyword and non-keyword to identify the keyword for each frame resulted in identifying the presence of the keyword in the whole utterance.

Query-by-Example (QbyE), is the third type of techniques developed for the development and evaluation of STD systems. QbyE is one of the earliest KWS methods [32], have two main steps including 1) *template representation* – how audio files of *needle(s)* are to be represented (e.g. in the form of lattices or posteriorgram feature vectors, etc.), and 2) *template matching* – how needles are to be matched with the *haystack* to find the desired *needle*. Various research efforts have been over the past decades [28] [33] [34] for unique *template representation* methods and variants of Dynamic Time Warping (DTW) [20] used for *template matching* phase [26].

The recent resurgence of *Neural Networks* (NN) as *Deep Neural Network* (DNN) gives a high rise to the KWS research area. Recently, Abdulkader et.al [1] proposed a model for KWS in narrowband audio, for computationally constrained devices by making use of DNNs, cascading, multiple-feature representations, and multiple-instance learning. In order to reduce the rate of false positives, they trained two classifiers on two different representations, Mel Frequency Cepstral Coefficient (MFCC) and Perceptual Linear Prediction (PLP) features. Moreover, Chen et.al [3] proposed a novel QbyE-STD method using Long Short-Term Memory (LSTM) based feature extractor. They showed that their presented KWS approach has low computation cost with high precision, can be efficiently used for small computational power devices.

Although, all of the above described methods shown to be very effective for the KWS task, assume the availability of large quantities of labeled speech data for training and testing of complex statistical language and acoustic models. For instance, one major drawback of LVCSR based KWS systems is Out-Of-Vocabulary (OOV) words, which is the main reason that LVCSR based methods best performed for well-resourced languages [30][39][40]. Similarly, Keyword/Filler based methods require prior knowledge of keywords and non-keyword elements to build special decoding graphs [3]. Chen et.al [3], to train DNN based KWS system, 19,000 audio files from 200 individuals used as positive examples whereas for negative examples a repository of audio samples of various meeting recordings were used. QbyE techniques normally take thousands of examples of *needles,* decoded by using ASR to acquire their lattice representation as templates and make detection decisions by comparing them against the *haystacks*. Moreover, the available techniques are computationally expensive due to their base on ASR. Therefore, these KWS methods were not suitable in low-resource contexts, and the reasons commercial firms focus on a few hundred languages of the world.

Transcription of the speech files, a major barrier in producing resources for under-resourced languages because it is not only an expensive process but also a time-consuming task. To address this challenge, the area of zero-resource or unsupervised speech processing emerged, by extracting meaningful features and learning language structures directly from unlabeled raw speech data [9][11][26][33][35]. With the advancements of Internet and multimedia technologies, it is quite easy to get audio data without transcription which makes it possible to develop speech processing solutions for under-resourced languages such as Urdu.

In the past, there are limited research efforts for the development of KWS for Urdu language. Irtza et.al [17] reported a KWS for Urdu language using filler modeling to compute non-keyword elements. A *phoneme recognizer* (PR) was used to model all phones. The audio input file is processed using PR and KWS, an achieved overall accuracy of 94.59%. Another work [45] also has been carried out for Urdu KWS task, but was developed only for five words of Urdu and achieved an accuracy level of 98.1%.

As far as our background knowledge and literature review, currently there is no completely unsupervised publically available KWS system developed for Urdu language because there are very limited standardized audio dataset which can be used for the development and evaluation of the KWS for Urdu language. Keeping in view the high demand of Urdu KWS system, we have developed a completely unsupervised QbyE-STD system (using the baseline approach proposed in [26]) by using the currently limited available gold-standard resources [37].

3. Methodology

We have developed an unsupervised STD system for Urdu that output all the occurrences of a *needle* (Q) in a given *haystack* (X), provided the input keyword (i.e. *needle*) is already present in the reference audio file (i.e. *haystack*). Our approach is most similar to the one proposed in [26] with the difference that we have used this approach and tune the parameters for Urdu language which is more phonetically rich than the English. Instead of using any phoneme recognizer, raw speech files were modeled using a Gaussian Mixture Model (GMM) without any supervision and get Gaussian Posteriorgram (GP). Segmental DTW (SDTW) was used to compare the distance between *Q* and *X* and generate the list of minimum distances in descending order.

Figure 1 illustrates the abstract level architecture of the developed KWS for Urdu. The acoustic model was trained, resulted in GPs of the training data. GMM was applied to get GPs of both of the *Needle* Q and *haystack* X. SDTW window was moved over the X and get occurrences (x1, x2...) of Q in X. This task is done without doing any explicit speech recognition.

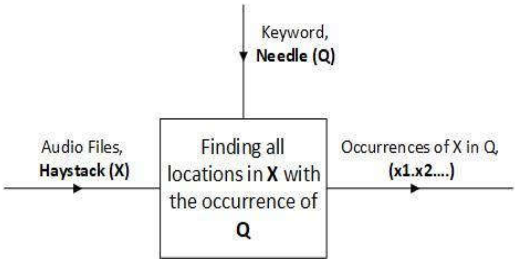

Figure 1: High-level architecture of the system.

3.1. Gaussian Mixture Model (GMM)

Posteriorgram is basically a probability vector which is used to represent the probabilities of the Gaussian components in a given speech frame. It is mostly used in the phonetic posteriorgram. Formally, if we represent speech by *n* frames:

$$S = (s_1, s_2, \dots, s_n) \quad (1)$$

The Gaussian probability vector is defined as in [26]:

$$GP(S) = (q_1, q_2, \dots, q_n) \quad (2)$$

Figure 2 demonstrates the process of computing GP vectors of both Q and X. Acoustic model was obtained by applying GMM on each frame of each audio file in the training data, to get a raw GP vector of each frame. This becomes a critical task when you do not have any labeled data. As reported in [26], training was performed by assuming that there are the same labels on all frames of the dataset which induces a problem of not discriminating between phonetic units in the posteriorgrams vector. Probability distribution on a large mass is concentrated on some dimension and the remaining dimensions have very little probability. To solve this problem, a speech/non-speech detector was applied to the training data by extracting the MFCC's and then GMM on them.

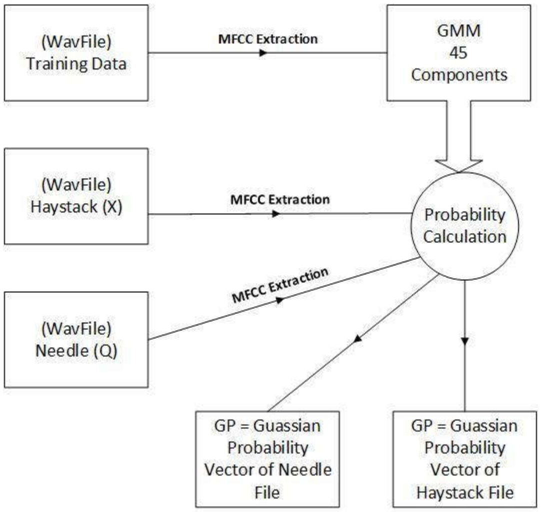

Figure 2: Computing Gaussian Posteriorgrams Vectors using GMM

Each element in GP(s) is representing a vector which can be calculated by using GMM. For example any q_j = $(P(c_1js_i),P(c_2js_i),$−−−$,P(c_mjs_i),)$, where c is representing the components of GMM and m is the size of Gaussian components. In this case, there are 45 Gaussian components which clustered the training data into 45 clusters. For *needle* Q and *haystack* X, probabilities were computed with respect to 45 clusters resulted in the probability vector of size 45 for each frame. Hence, the GP matrix *M* size is *number of frames time's Gaussian components (Matrix Size* (M) *= No. of frames ∗ Gaussian components)*

For both audio files of *needle* Q and *haystack* X, each speech file divided into windows *aka* frames of 25 *msec* along with the overlapping step size of 10 msec, to avoid missing any information of the signal at the window boundaries. For each frame, the probability vector of size 45 was obtained by passing it through the GMM processor to make it GP vector. Figure 3 provides the visualization of *needle* Q framing and its respective GP vector with 45 GP elements. Similarly, the probability vector of size 45 for *haystack* X will be computed, and the visual representation of file *X* is similar to *Q* file.

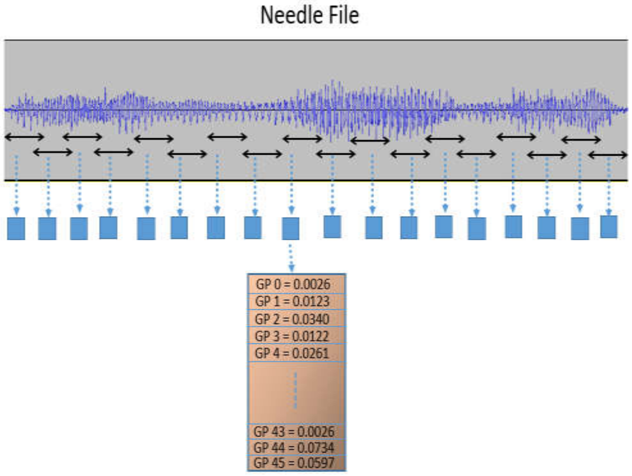

Figure 3: Visualization of needle file.

3.2. Segmental Dynamic Time Warping (SDTW).

SDTW is the modified version of well renowned DTW algorithm [20], and has demonstrated its success in unsupervised pattern discovery in audio files [26] [45] [47]. It works by finding the distance between the elements of both (*needle* and *haystack*) signals and then finds a path with minimum distance between these elements. To find Q in X, SDTW was applied to GP vectors of both Q and X. The distance between two GP vectors computed using equation (3):

$$D = -log(p.q) \qquad (3)$$

Where p and q are two posteriorgram vectors. As both *p* and *q* are probability vectors, dot product was used as a similarity measure to find the distance between them. By applying SDTW, there is a need to handle the following two constraints: 1) Adjustment window condition and 2) the step length of the start coordinate of the DTW search [26]. Fixation of the adjustment window size will restrict the shape and ending coordinate of the warping path, but if use different starting coordinates then the warping path will be automatically in the diagonal regions of the DTW grid. Therefore, we used overlapping window strategy and every time move window (adjustment window size) *R* steps for the next search. The reason for using the overlapping window is to avoid redundant computation and to check the warping path across the boundary of segments. Size of *Q* is fixed in this case and just need to care about the segments of the *X*. Window will be moved R steps forward in *X* and *no of warping path* (by equation (4)) will come as an outcome, where each path represents the warping between *Q* and *X*.

$$\frac{(n-1)}{R} \tag{4}$$

3.3 System Flow

Figure 5 demonstrates the flow of the reported Urdu KWS system. The steps are as follows:

1. Raw input speech of both *Q* and *X* given to the system.
2. Remove the silence from Q and X by using Voice Activity Detector (VAD), because while comparing *Q* with the frames of X, the silence was also compared ended up in false results.
3. MFCC (i.e. 13 coefficients) vectors are extracted from Q, X and training data.
4. GMM is applied to the MFCC vectors of training data (audio file of about 1 hour speech) to make the optimum number of phonetic clusters.
5. GP vectors (as shown in Figure 3) of MFCCs are calculated for both *Q* and *X*.
6. By taking overlapping frames from the GP vector of *Q* and *X*, SDTW is applied using the dot product (cosine similarity [15]) as the distance measuring method.
7. Results are sorted in ascending order of cost.

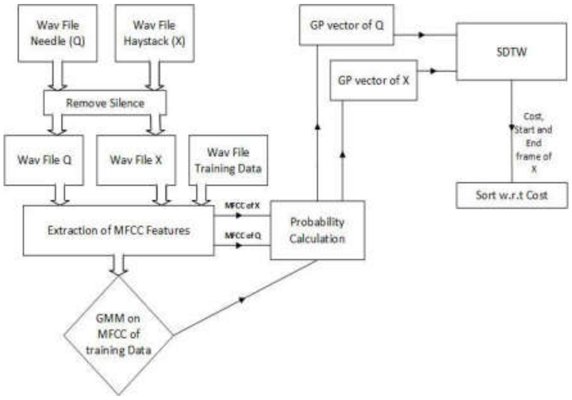

Figure 5: The system flow diagram

4. Experimental Setup

4.1. Types of Experiments

Two types of experiments were performed including 1) **Same Speaker** – training and testing audio files are in the voice of the same speaker, and 2) **Cross Speaker** – training audio files speaker is different from the test audio files speaker. For the same speaker experiments, 15 words (i.e. *needles*) were selected whereas for cross speaker experiment, 2 words were selected.

4.2. Dataset

For the development and evaluation of the proposed KWS system for Urdu language, PRUS [20] corpus was used. It is not as larger as the other available benchmark speech corpora for English (e.g. TIMIT [48], Librispeech [49], etc.), but for Urdu it is the only publically phonetically rich (covers almost all of the Urdu language sound) gold standard corpus. It contains 708 audio files in .wav format, in total 90 minutes of Urdu speech.

4.3. Training and Testing Data

For the same speaker experiments: for each needle, all of the audio files in the dataset were used for training of the GMM except for those files contained the selected needle in the respective experiment. For instance, the word "پاس" (occurred in 8 audio files out of 708) is selected as a *needle* in experiment number 1. GMM will be trained in those 700 audio files which do not contain the selected *needle*. Now, out of the 8 files having the *needle* in each file, 7 files will be selected as *haystack* files whereas *needle* word will be extracted from the remaining 1 audio file.

For cross speaker experiments: all of the 708 audio files were used to train GMM, whereas 2 words were recorded from another speaker as a *needle*.

4.4. Evaluation Measures

To evaluate the developed unsupervised KWS for Urdu language, Precision (P) was used as an evaluation measure because of the constraint "needle(s) must be present in the haystack(s)". This implies that only true positives and false positives can be computed for the proposed system. We have P@N [8]: the precision of the top N hits, where N is the number of occurrences of a needle in the haystack.

5. Results and Analysis

The summarized results of the experiments for same and cross speaker are shown in Table 1 and Table 2 respectively, where N shows the number of needles' occurrences, P@N shows the precision of finding N number of Q instances in X, and P@3 represents the precision of locating Q in those X files which contains 3 occurrences of Q. Similarly P@5 and P@10 indicates precision of respective X files. Cells with value 'NA' show that there is no X available with the required number of occurrences for that specific Q. The last row in both tables shows the average precision of all experiments.

It is clear from Table 1 that the average P@3 (82.30 %) outperforms whereas the mean P@10 (76 %) performed worse than all other. It can also be seen that the average P@N (79.20 %) is comparative to that of P@5 (80 %). It is clear from the average precision results and individual keyword results that the precision decreases as the number of occurrences of a needle increases. Another possible reason for this precision degradation could be the middle vowels, as GMM performed best on 45 phonetic clusters although there exist 67 different phonemes in Urdu language.

Table 1: Summarized experimental result for same speaker.

Needle	N	P@N	P@3	P@5	P@10
پاس	8	0.72	0.83	0.83	NA
ایک	28	0.7	0.75	0.6	0.55
بعد	26	0.9	1	1	0.71
نہیں	15	0.71	0.42	0.5	0.62
جاتا	6	1	1	1	NA
صاحب	6	0.75	0.75	0.71	NA
غیر	5	0.6	0.625	0.6	NA
ساتھ	23	0.85	1	1	0.9
والے	6	0.75	1	0.55	NA
پہلے	6	0.6	0.5	0.71	NA
کیلئے	14	0.73	0.75	0.71	1
جانے	6	1	1	1	NA
کرنے	11	0.73	0.75	1	0.71
سامنے	5	1	1	1	NA
بغیر	10	0.83	1	0.83	0.83
Avg. Precision		**79.20%**	**82.30%**	**80%**	**76%**

The cross speaker experimental results are shown in Table 2. It can be seen that the average P@5 (100%) outcompeted all other average precisions (i.e. 91.5 %). The proposed system located the needle "قسطنطنیہ" perfectly in all settings. The possible reason could be the uniqueness of the needle due to its larger phonemic counts, quite unique phonetic sequence, and this word is not commonly used in conversational Urdu speech. Whereas the needle "بغداد" correctly spotted only when the number of occurrences is 5 while in other settings the performance is decreased. The shorter phonemic length and common phonetic sequence could be the probable reasons for this low precision.

The proposed KWS performed better in cross speaker settings as compared to the same speaker. One obvious reason could be the number of needles chosen for the experiments, which are too less in cross speaker scenario. Another important observation for this low precision in same speaker context, could be the length of phonemes in each needle as in same speaker experiments the average needle length is 3 whereas in cross speaker settings it is 9 which implies that needles with shorter phonemic counts are harder to locate as compared to the needles with larger phonemic count.

It has also been found that the produced results seem to strongly dependent upon the type (unique) of words. Words present as substring may increase false-positive results. For example, the word "Tania" and "Aania" are almost the same because "Aania" is present as a substring. Our system saying these words are the same and that is not true. As far as our domain knowledge and literature review, this is the first attempt to develop an Urdu KWS system in a completely unsupervised manner. The initial results demonstrate that there is still a big room available to improve Urdu KWS.

Table 2: Summary of results of cross speaker experiments

Needle	N	P@N	P@3	P@5	P@10
قسطنطنیہ	10	1	1	1	1
بغداد	10	0.83	0.83	1	0.83
Avg. Precision		**91.5%**	**91.50%**	**100%**	**91.50%**

6. Conclusion and Future Work

Availability of the large datasets is a crucial requirement for the majority of the existing KWS techniques as they require huge datasets to train the model, which makes these methods unsuitable for low resource languages like Urdu. Keeping in view the high demand for the KWS system for Urdu, this paper reports an unsupervised STD system for Urdu.

Without any transcription, the model is trained by extracting MFCCs directly from speech files. GMM is applied to the training data to make the phonetic clusters, and generate GPs for both of the needle and haystack. Segmental DTW, a modified version of the well renowned DTW signal alignment method, used to compare the GP vectors of the input keyword and the reference audio file. Warping path with minimum score indicates the frames associated with this path are closer to each other. Experiments were performed for both same and cross speaker settings, and observed that the proposed system performed better in cross speaker scenarios as compared to the same speaker context.

The proposed system has some limitations such as 1) the major constraint "given the word is present in haystack", 2) it reports all the occurrences of a needle in a given haystack but, it does not tell either the word is present or not, and 3) length normalization of vectors because DTW returns different scores for different lengths of vectors. To overcome all of these limitations is the future goal of this work to obtain more satisfactory results along with examining this method on other low resource regional languages such as Pashto or Punjabi.

7. Conclusion and Future Work

The code and the dataset described in this article are freely available for research purposes and can be downloaded from https://github.com/ab-101/Key-Word-Spotter.

8. Bibliographical References

[1] Abdulkader, Ahmad, Kareem Nassar, Mohamed Mahmoud, Daniel Galvez, and Chetan Patil. "Multiple-Instance, Cascaded Classification for Keyword Spotting in Narrow-Band Audio." arXiv preprint arXiv:1711.08058 (2017).

[2] C. Richard C. A. Rose and B. Douglas. “A Hidden Markov Model Based Keyword Recognition System”. In *Acoustics, Speech, and Signal Processing, ICASSP-90,*, pages 129–132, IEEE, 1990.

[3] G. Chen, C. Parada, and T. N. Sainath. “Query-by Example Keyword Spotting using Long Short-Term Memory Networks”. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5236–5240, April 2015.

[4] S. Cox and R. Rose. “A confidence measures for theˆ switchboard database. In *Proceedings of ICASSP*, volume 1, pages 511–515, 1996.

[5] S. Das. ”speaker dependent bengali keyword spotting in unconstrained english speech”. 2005.

[6] Samarjit Das. Speaker dependent Bengali keyword spotting in unconstrained English speech acknowledgement. 2005.

[7] B. Samy H. Ketabdar, V. Jithendra and B. Herve. Posterior based keyword spotting with a priori thresholds. In *Ninth International Conference on Spoken Language Processing*, 2006.

[8] T. J. Hazen, W. Shen, and C. White. Query-by-example spoken term detection using phonetic posteriorgram templates. In *2009 IEEE Workshop on Automatic Speech Recognition Understanding*, pages 421–426, Nov 2009.

[9] M. Huijbregts, M. McLaren, and D. van Leeuwen. Unsupervised acoustic sub-word unit detection for queryby-example spoken term detection. In *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4436–4439, May 2011.

[10] P. Matejka L. Burget M. KarafiA¡t I. Szoke, P. Schwarz˜ and J. Cernocky. Phoneme based acoustics keyword spotting in informal continuous speech. In *International Conference on Text, Speech and Dialogue*, pages 302–309, Berlin, 2005.

[11] Aren Jansen and Kenneth Church. Towards unsupervised training of speaker independent acoustic models. pages 1693–1692, 01 2011.

[12] Jochen Junkawitsch, L. Neubauer, Harald Hoge, and¨ Gunther Ruske. A new keyword spotting algorithm¨ with pre-calculated optimal thresholds. *Proceeding of Fourth International Conference on Spoken Language Processing. ICSLP ’96*, 4:2067–2070 vol.4, 1996.

[13] Jochen Junkawitsch, L. Neubauer, Harald Hoge, and¨ Gunther Ruske. A new keyword spotting algorithm¨ with pre-calculated optimal thresholds. *Proceeding of Fourth International Conference on Spoken Language Processing. ICSLP ’96*, 4:2067–2070 vol.4, 1996.

[14] Hamed Ketabdar, Jithendra Vepa, Samy Bengio, and Herve Bourlard. Posterior based keyword spotting with a priori thresholds. 01 2006.

[15] R. Dehak P. Dumouchel N. Dehak, P. Kenny and P. Ouellet. Front-end factor analysis for speaker verification. *Trans. Audio, Speech, Lang. Process., vol. 19, no. 4*, pages 788–798, IEEE 2011.

[16] R. Rose and D. B. Paul. A hidden markov model based keyword recognition system. In *Acoustics, Speech, and Signal Processing ICASSP-90*, pages 129–132, IEEE, 1990.

[17] Irtza, S., Rehman, K., and Hussain S. ”Urdu Keyword Spotting System using HMM,”. 2012.

[18] I. Zaharakis S. Kotsiantis, Sotiris B. and P. Pintelas. Emerging artificial intelligence applications in computer engineering 160. In *Supervised machine learning: A review of classification techniques*, pages 3–24. 2007.

[19] T. Hain D. Kershaw G. Moore J. Odell D. Ollason D Povey V. Valtchev S. Young, G. Evermann and P. Woodland. *HTK Book*. Microsoft Corporation and Cambridge University Engineering Department, 3.2.1 edition, 2002.

[20] H. Sakoe and S. Chiba. Dynamic programming algorithm optimization for spoken word recognition. In *Proceedings of Trans. on Acoustic Speech, and Signal Processing ASSP 26*, pages 43–49, IEEE 1978.

[21] Wade Shen, Christopher M. White, and Timothy J. Hazen. A comparison of query-by-example methods for spoken term detection. In *INTERSPEECH*, 2009.

[22] Gyorgy Szasz¨ ak and Andr´ as Beke. Using phonological´ phrase segmentation to improve automatic keyword spotting for the highly agglutinating hungarian language. In *INTERSPEECH*, 2013.

[23] Igor Szoke, Petr Schwarz, Pavel Matejka, Lukas Burget, Martin Karafiat, Michal Fapso, and Jan Cernocky. Comparison of keyword spotting approaches for informal continuous speech. pages 633–636, 01 2005.

[24] T. Kemp T. Schaaf. Confidence measures for spontaneous speech recognition. In *Proceedings of ICASSP*, volume 2, pages 887–890, 1997.

[25] Javier Tejedor and JosA˜ c ColA¡s. Spanish keyword˜ spotting system based on filler models, pseudo n-gram language model and a confidence measure. 01 2006.

[26] Z.Yaodong and J. R. Glass. Unsupervised spoken keyword spotting via segmental dtw on gaussian posteriorgrams. In *Automatic Speech Recognition Understanding, IEEE workshop on ASRU*, pages 398–403, 2009.

[27] https://www.coursera.org/

[28] Ao, Chia-Wei, and Hung-yi Lee. "Query-by-example spoken term detection using attention-based multi-hop networks." In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6264-6268. IEEE, 2018.

[29] Chung, Yu-An, and James Glass. "Speech2vec: A sequence-to-sequence framework for learning word embeddings from speech." arXiv preprint arXiv:1803.08976 (2018).

[30] Chen, Guoguo. "Low Resource High Accuracy Keyword Spotting." PhD diss., Johns Hopkins University, 2016.

[31] https://en.wikipedia.org/wiki/Languages_of_South_Asia

[32] Bridle, John S. "An efficient elastic-template method for detecting given words in running speech." In Brit. Acoust. Soc. Meeting, pp. 1-4. 1973.

[33] Huijbregts, Marijn, Mitchell McLaren, and David Van Leeuwen. "Unsupervised acoustic sub-word unit detection for query-by-example spoken term detection." In 2011 IEEE international conference on Acoustics, speech and signal processing (ICASSP), pp. 4436-4439. IEEE, 2011.

[34] Shen, Wade, Christopher M. White, and Timothy J. Hazen. A comparison of query-by-example methods for spoken term detection. MASSACHUSETTS INST OF TECH LEXINGTON LINCOLN LAB, 2009.

[35] Wang, Haipeng, Tan Lee, and Cheung-Chi Leung. "Unsupervised spoken term detection with acoustic segment model." In 2011 International Conference on Speech Database and Assessments (Oriental COCOSDA), pp. 106-111. IEEE, 2011.

[36] Zhang, Yaodong, and James R. Glass. "Unsupervised spoken keyword spotting via segmental DTW on Gaussian posteriorgrams." In 2009 IEEE Workshop on Automatic Speech Recognition & Understanding, pp. 398-403. IEEE, 2009.

[37] Raza, Agha Ali, Sarmad Hussain, Huda Sarfraz, Inam Ullah, and Zahid Sarfraz. "Design and development of phonetically rich Urdu speech corpus." In 2009 oriental COCOSDA international conference on speech database and assessments, pp. 38-43. IEEE, 2009.

[38] C. Chelba, T. Hazen and M. Sarac¸lar, "Retrieval and browsing of spoken content," IEEE Signal Processing Magazine, vol. 24, no. 3, pp. 39–49, May 2008.

[39] D. Miller, et al, "Rapid and accurate spoken term detection," in Proc. Interspeech, Antwerp, Belgium, 2007.

[40] J. S. Garofolo, C. G. Auzanne, and E. M. Voorhees, "The TREC spoken document retrieval track: a success story," NIST SPECIAL PUBLICATION SP, no. 246, pp. 107–130, 2000.

[41] M. Sarac¸lar and R. Sproat, "Lattice-based search for spoken utterance retrieval," in Proc. HLT-NAACL, Boston, 2004.

[42] A. Mandal, K. P. Kumar, and P. Mitra, "Recent developments in spoken term detection: a survey," International Journal of Speech Technology, vol. 17, no. 2, pp. 183–198, 2014.

[43] R. C. Rose and D. B. Paul, "A hidden Markov model based keyword recognition system," in Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP). IEEE, 1990, pp. 129–132.

[44] J. Wilpon, L. Miller, and P. Modi, "Improvements and applications for key word recognition using hidden Markov modeling techniques," in Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP). IEEE, 1991, pp. 309–312.

[45] Juang, Biing-Hwang. "Recent developments in speech recognition under adverse conditions." In First International Conference on Spoken Language Processing. 1990.

[46] A. Park and J. Glass,"Unsupervised pattern discovery in speech", in IEEE Trans. ASLP, 6(1), 1558–1569, 2008.

[47] Chan, Chun-an, and Lin-shan Lee. "Unsupervised spoken-term detection with spoken queries using segment-based dynamic time warping." In Eleventh Annual Conference of the International Speech Communication Association. 2010.

[48] Garofolo, John S. "TIMIT acoustic phonetic continuous speech corpus." Linguistic Data Consortium, 1993 (1993).

[49] Panayotov, Vassil, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. "Librispeech: an ASR corpus based on public domain audio books." In 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 5206-5210. IEEE, 2015.