Unsupervised Speech Unit Discovery Using K-means and Neural Networks

Céline Manenti(✉), Thomas Pellegrini, and Julien Pinquier

IRIT, Université de Toulouse, UPS, Toulouse, France
{celine.manenti,thomas.pellegrini,julien.pinquier}@irit.fr

Abstract. Unsupervised discovery of sub-lexical units in speech is a problem that currently interests speech researchers. In this paper, we report experiments in which we use phone segmentation followed by clustering the segments together using k-means and a Convolutional Neural Network. We thus obtain an annotation of the corpus in pseudo-phones, which then allows us to find pseudo-words. We compare the results for two different segmentations: manual and automatic. To check the portability of our approach, we compare the results for three different languages (English, French and Xitsonga). The originality of our work lies in the use of neural networks in an unsupervised way that differ from the common method for unsupervised speech unit discovery based on auto-encoders. With the Xitsonga corpus, for instance, with manual and automatic segmentations, we were able to obtain 46% and 42% purity scores, respectively, at phone-level with 30 pseudo-phones. Based on the inferred pseudo-phones, we discovered about 200 pseudo-words.

Keywords: Neural representation of speech and language · Unsupervised learning · Speech unit discovery · Neural network · Sub-lexical units · Phone clustering

1 Introduction

Annotated speech data abound for the most widely spoken languages, but the vast majority of languages or dialects is few endowed with manual annotations. To overcome this problem, unsupervised discovery of linguistic pseudo-units in continuous speech is gaining momentum in the recent years, encouraged for example by initiatives such as the *Zero Resource Speech Challenge* [15].

We are interested in discovering pseudo-units in speech without supervision at phone level ("pseudo-phones") and at word level ("pseudo-word"). Pseudo-words are defined by one or more speech segments representing the same phonetic sequence. These are not necessarily words: they may not start/end at the beginning/ending of a true word and may contain several words, such as "I think that". The same applies for pseudo-phones that may be shorter or longer than true phones, and one pseudo-phone may represent several phones.

To find speech units, one can use dotplots [4], a graphical method for comparing sequences, and Segmental Dynamic Time Warping (S-DTW) with the

DOI: 10.1007/978-3-319-68456-7 14

use of the cosine similarity that gives distances between phonetic acoustic models [1,11]. In [17], DTW and Hidden Markov Models are also used on posteriorgrams (posterior distribution over categorical units (e.g. phonemes) as a function of time) to find pseudo-words.

During our own experiments, we were able to see the usefulness of the posteriorgrams, which are data obtained by supervised learning. We therefore sought to obtain these posteriorgrams phones in an unsupervised way.

To obtain phone posteriorgrams, clustering can be used. In [17], k-means are used on parameters generated by an auto-encoder (AE), also called Bottleneck Features (BnF), after binarization. k-means are similarly used in [14], with AEs and graph clustering. Increasingly used in speech research, neural networks come in several unsupervised flavors. AEs learn to retrieve the input data after several transformations performed by neuron layers. The interesting parameters lie in the hidden layers. AEs can have several uses: denoising with the so-called denoising AEs [16], or creating new feature representations, such as Bottleneck features using a hidden layer with a number of neurons that is markedly lower than that of the other layers. The information is reformulated in a condensed form and the AE is expected to capture the most salient features of the training data. Studies have shown that, in some cases, AE posteriorgrams results are better than those of GMM [2,7]. In the context of unsupervised speech unit discovery, AE variants have emerged, such as correspondence AEs (cAEs) [13]. cAEs no longer seek to reconstruct the input data but other data, previously mapped in a certain way. They therefore require a first step of grouping segments of speech into similar pairs (pseudo-words, etc.) found by a DTW. There is another type of AEs, which avoids the DTW step by forcing to reconstruct neighboring data frames, using the speech stability properties: the so-called segmental AEs [2].

In our work, we first performed tests with different AEs. Their results mainly helped to separate the voiced sounds from the unvoiced sounds but gave poor results for our task of pseudo-phone discovery (less than 30% purity on the BUCKEYE corpus, see Sect. 3). We decided to design an alternative approach coupling k-means and supervised neural networks.

This paper is organized as follows. Section 2 presents the system architecture, then the speech material used to validate our approach on three languages is described. Finally, results both at phone- and word-levels are reported and discussed in Sect. 4.

2 System Description

Figure 1 shows the schema of the system. First, pseudo-phones are discovered by a k-means algorithm using log F-banks as input. Second, a CNN classifier is trained to predict these pseudo-units taken as pseudo-groundtruth. The probabilities outputted by the CNN are then used as features to run k-means once again. These last two steps (supervised CNN and k-means) are iterated as long as the CNN training cost decreases.

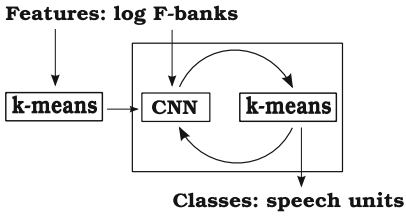

Fig. 1. General architecture: the system is trained in an iterative manner

2.1 Features

As input, we use 40 log Filterbank coefficients extracted on 16-ms duration windows, with a 3/4 hop size. 40 Mel-filters is a conventional number of filters used for speech analysis tasks.

Currently, our model needs to know the boundaries of the phones in order to standardize the input feature at segment level. We use and compare two different segmentations: the manual segmentation provided with the corpus and a segmentation derived automatically based on our previous work on cross-language automatic phone-level segmentation [8].

2.2 Class Assignment for CNN Learning

For initialization, the k-means algorithm uses frames of log F-bank coefficients as input and each input feature window is concatenated with its 6 neighborhood windows. It assigns a single class per window and we propagate this result on the segments delimited by the phone boundaries by a majority vote. Figure 2 illustrates the majority voting strategy used to choose the single pseudo-phone number 7 on a given segment. In the following iterations, the k-means algorithm takes as input the phone posteriorgrams generated by the CNN.

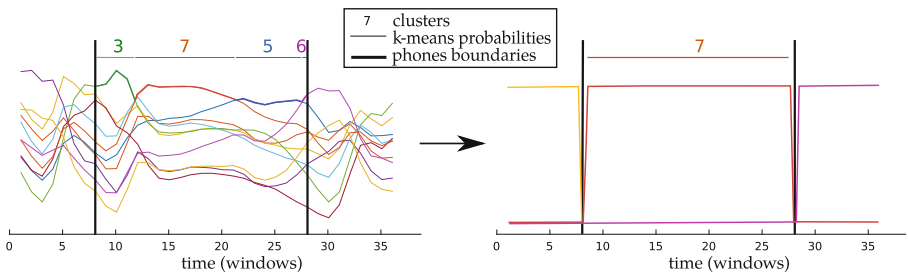

Fig. 2. Majority voting strategy

As we require the CNN to output the same class for all the windows comprising a segment, the model learns to output rather stable probabilities on the

segments. We therefore accelerate the k-means step by using as input a single window, which is the average of the windows comprising a given segment and we obtain directly a class by segment. This simplification was shown in preliminary experiments to have no impact on the results. As a result, we have a single class assigned by segment.

2.3 CNN Architecture

Supervised neural networks require to know the classes of the training data. In our case, the true manually annotated phones are not available, thus, we use pseudo-phones clusters previously inferred by the k-means algorithm based on a previous segmentation of the input data.

We use a CNN with two convolution layers followed by a fully connected layer and a final output layer. The nonlinearity function used is the hyperbolic tangent. The first convolution layer is comprised of thirty 4×3 filters followed by a layer of 2×2 maxpooling and the second one of sixty 3×3 filters followed by a layer of 1×2 maxpooling. The dense layer has 60 neurons and we use dropout (0.5) before the last layer. A 0.007 learning rate was used with Nesterov momentum parameter updates.

Our experiments showed that the iterative process using pseudo-phones inferred by the k-means algorithm gives better results than those attributed by the first k-means iteration. Moreover, the CNN can also give us for each window the class probabilities. After experiments, these posteriorgrams outperformed the F-bank coefficients. We have chosen to retain these probabilities, on which we apply the consecutive k-means iterations. Our model is therefore an iterative model.

3 Speech Material

For our experiments, we used three corpus of different languages, sizes and conditions.

3.1 BUCKEYE

We used the American English corpus called BUCKEYE [12], composed of spontaneous speech (radio recordings) collected from 40 different speakers with about 30 min of time speech per speaker. This corpus is described in detail in [6].

The median duration of phonemes is about 70 ms, with 60 different phonemes annotated. It is more than the 40 usually reported for English, because of peculiar pronunciations that the authors of BUCKEYE chose to distinguish in different classes, particularly for nasal sounds.

We used 13 h of recordings of 26 different speakers, corresponding to the part of training according to the subsets defined in the *Zero Resource Speech* 2015 challenge [15].

3.2 BREF80

BREF80 is a corpus of read speech in French. As we are interested in less-resourced languages, we only took one hour of speech, recorded by eight different speakers. The French phone set we considered is the standard one comprised of 35 different phones, with a median duration of 70 ms.

3.3 NCHLT

The Xitsonga corpus [5], called NCHLT, is composed of short read sentences recorded on smart-phones, outdoors. We used nearly 500 phrases, with a total of 10,000 examples of phonemes annotated manually, from the same challenge database than the one used in the *Zero Resource Speech challenge*. The median duration of the phones is about 90 ms and there are 49 different phones.

4 Experiments and Results

In this section, we first report results with manual phone segmentations in order to evaluate our approach on the pseudo-phone discovery task only. Results at phone and word levels are given. In the last Subsect. 4.3, we evaluate the system under real conditions, namely with our automatic phone-level segmentations. We evaluate our system on different languages and speaking styles.

4.1 Results at Phone Level

To evaluate our results, we compute the standard purity metric of the pseudo-phones [9].

Let N be the number of manual segments at phone level, K the number of pseudo-phones, C the number of phones and n_j^i the number of segments labeled with phone j and automatically assigned to pseudo-phone i. Then, the clustering purity obtained is defined as:

$$\frac{1}{N}\sum_{i=1}^{K}\arg\max_{j\in[1,C]}(n_j^i)$$

First, we sought to optimize the results of the first k-means' iteration, the one used to initialize the process by assigning class numbers to segments for the first time. The parameters that influence its results are the input features (log F-bank coefficients), the context size in number of frames and the number of means used.

We tested different context sizes and found that the influence of this parameter was at most one percent on the results. The best value is around six windows.

The choice of the number of clusters is ideally in the vicinity of the number of phones sought, that is to say generally about thirty. This is an average value

of course, and there are languages that comprise many more phones, such as the Khoisan language with 141 different phones. We will look at the influence of the mean number on the search for pseudo-words in Sect. 4.2.

It is interesting to compare the results with the ones obtained in a supervised learning setting. Table 1 shows the results in terms of purity. As expected, results obtained with the supervised CNN are much better than with the clustering approach. Figure 3 shows the improvement provided by the use of a neural network. It is for a small number of pseudo-phones that the CNN improves the results the most (almost + 10% for 15 clusters).

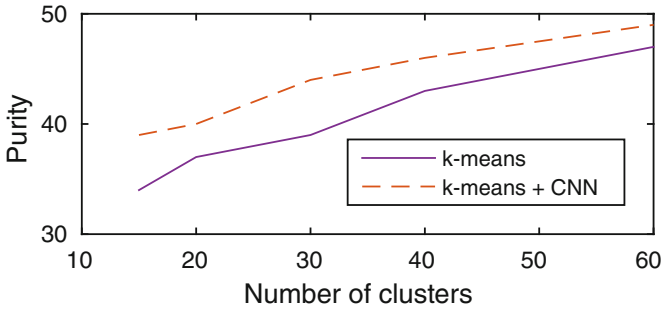

Fig. 3. Improvement in percent purity thanks to the neural network

One of the possible applications of this work is to help manual corpus annotation. In the case where a human would label each of the clusters attributed by the model with the real phonetic labels, thus regrouping the duplicates, we can consider using more averages than the number of phones present in the language considered. We have therefore looked at the evolution of purity as a function of the number of clusters in Fig. 4.

We see that, for few clusters, the results improve rapidly. But, starting from a hundred clusters, purity begins to evolve more slowly: in order to gain about 4% in purity, the number of averages needs to be multiplied by a factor of 10.

Table 1 gives the following pieces of information to evaluate the quality of the results:

- The percentage of purity obtained in supervised classification by the same CNN model as the one we use in the unsupervised setting. We did not try to build a complicated model to maximize the scores but rather a model adapted to our unsupervised problem. In comparison, the state of the art is around 80% of phone accuracy on the corpus TIMIT [3].
- The percentage of purity obtained by our unsupervised model. Scores are calculated for 30 pseudo-phones. We see that there are almost 20% of difference between the small corpus of read Xitsonga and the large spontaneous English corpus.

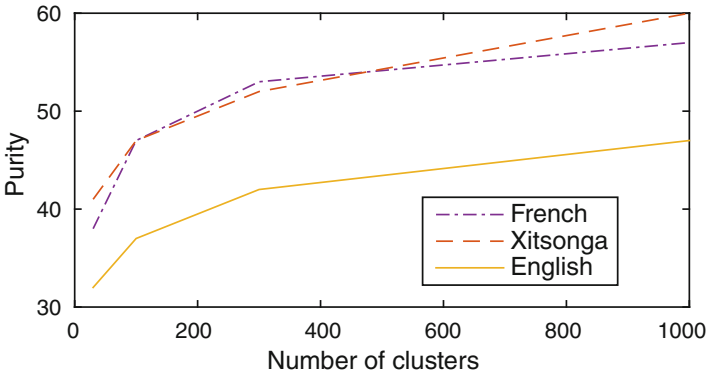

Fig. 4. Influence of the number of clusters on purity

– To further evaluate these results, we assign to each group the phonetic class most present among the grouped segments. Several groups can thus correspond to the same phone. The last line in the table indicates the number of different phones allocated for 30 groups found by our model. We see that we have only about fifteen different phones allocated, which is less than the number of phones present in these languages. So there are more than half of the phones that are not represented. This value changes slightly when we increase the number of clusters.

Table 1. Purity by segment (%) obtained for each corpus: English (En, BUCKEYE), French (Fr, BREF80) and Xitsonga (Xi, NCHLT)

Language	En	Fr	Xi
Purity (%) supervised learning	60	62	66
Purity (%) for 30 clusters	29	43	46
Number of $\neq$ phonetic classes	16	18	11

With the French and Xitsonga corpora, we obtained the best results, whether supervised or not. This can be easily explained: they are comprised of read speech, are the smallest corpora and with the least numbers of different speakers. These three criteria strongly influence the results. It is interesting to note that we get almost equivalent results with English if we only take 30 min of training data from a single speaker.

By studying in details the composition of the clusters, we found that having 30% (respectively 40%) purity scores does not mean that we have 70% (respectively 60%) of errors due to phones that differ from the phonetic label attributed during the clustering. The clusters are generally made up of two or three batches

of examples belonging to close phonetic classes. Thus, the three phones most frequent in each cluster represent on average 70% of their group samples for French or Xitsonga and 57% for English.

4.2 Results at Word Level

To find pseudo-words, we look for the sequences comprised of the same pseudo-phones. A pseudo-word must at least appear twice. We only consider sequences of more than 5 pseudo-phones. Using shorter pseudo-phone sequences leads to too many incorrect pseudo-words.

To evaluate these results, we compare the phone transcripts constituting the different realizations of a given pseudo-word. If these manual transcripts are identical, then the pseudo-word is considered as correct. Otherwise, we count the number of phone differences. For a pseudo-word with only two realizations, we accept up to two differences in their phone sequences. For a pseudo-word with more than two examples, we rely on a median pseudo-word as done in [10], and again tolerate two differences maximum.

The results may depend on the number of groups selected. If we consider a larger number of distinct clusters, we get less pseudo-phone sequences that are the same, and thus we discover less pseudo-words. But by doing so, the groupings are purer, as shown in the Fig. 5.

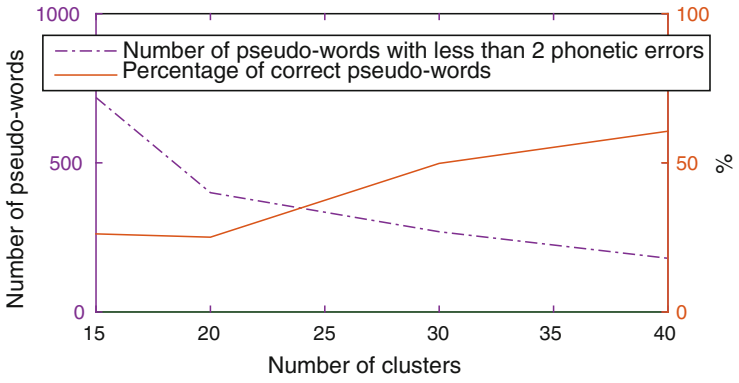

Fig. 5. Influence of the number of pseudo-phones on quantity and purity of pseudo-words found.

In Table 2, we look at three characteristics of the identified pseudo-words to evaluate our results:

- Number of pseudo-words found,
- Number of pseudo-words whose manual phonetic transcription between group examples has at most two differences,

Table 2. Pseudo-words statistics with manual and automatic segmentations

Language	En	Fr	Xi
# Hours	13	1	1/2
# Phone examples	586k	36k	10k
Manual segmentation			
# pseudo-words	3304	671	231
# Pseudo-words $\leq$ 2 differences	1171	415	172
# Identical pseudo-words	334	188	76
Automatic segmentation			
# Pseudo-words	3966	540	200
# Pseudo-words $\leq$ 2 differences	843	269	120
# Identical pseudo-words	40	32	25

– Number of pseudo-words in which all lists representing them have exactly the same phonetic transcription.

The French corpus allows to obtain 671 pseudo-words, out of which 188 are correct and 227 with one or two differences in their phonetic transcriptions. We thus find ourselves with 415 pseudo-words with at most two differences with their manual phonetic transcriptions of the examples defining them. The results obtained on the other two corpora are worse. Proportionally, we find about ten times less pseudo-words than for with the French corpus.

In comparison, in a work performed on four hours of the corpus ESTER, 1560 pseudo-words were found, out of which 672 of them were sufficiently accurate according to their criteria, with an optimized pseudo-word search algorithm based on the DTW and self-similarities [10].

4.3 Towards a Fully Automatic Approach: With Automatic Segmentations

Until now we have used manual phones segmentation. To deal with real conditions when working on a few resourced language, we will now use our model without input handwritten data. We therefore use automatic phones segmentation to train our model. This automatic segmentation can be learned in other languages, with more resources.

In a previous work, we used a CNN to perform automatic segmentation task [8]. It is a supervised model, but we have demonstrated its portability to languages other than those learned.

The CNN takes as input F-bank coefficients and outputs probabilities of the presence/absence of a boundary at frame-level. The diagram of this model is represented in the Fig. 6. The output is a probability curve evolving in time whose summits are the locations of probable boundaries. To avoid duplicates due

to noisy peaks, the curve is smoothed and all summits above a certain threshold are identified as phones boundaries.

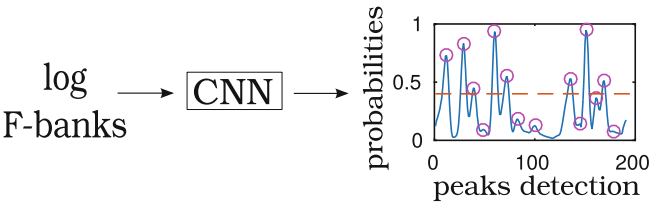

Fig. 6. Illustration of our automatic segmentation approach, based on a CNN

Previously, we showed that this network was portable to other languages than the one used for training and this is very useful in the present work. Indeed, for a language with few resources, we can use a corpus from other well-resourced languages. In addition, it is an additional step towards a fully unsupervised setting. Table 3 shows that the network gets a good F-measure on languages other than those used for training.

Table 3. F-measure (%) obtained with the automatic cross-language segmentation with the results on the columns according to the test corpus and on the lines according to the two training corpora, for 20 ms.

Languages	Test	En	Fr	Xi
Train	En + Fr	73	74	**53**
	Fr + Xi	**64**	80	64
	En + Xi	73	**63**	58

Concerning the pseudo-word discovery, we get the results displayed in Table 2. The number of pseudo-words found is similar to that found with the manual segmentation but the purity score is lower with this automatic segmentation, as expected. For French and Xitsonga, half of the pseudo-words found have less than two errors, for English it is less than a quarter.

5 Conclusions

In this paper, we reported our experiments on speech unit discovery based first on a simple approach using the k-means algorithm on acoustic features, second on an improved version, in which a CNN is trained on the pseudo-phones clusters inferred by k-means. This solution differs from the standard approach based on AEs reported in the literature.

Our model is not yet fully unsupervised: it needs a pre-segmentation at phone level and obviously the best results were obtained with a manual segmentation. Fortunately, the loss due to the use of automatic segmentation is small and we have shown in a previous work that this segmentation can be done using a segmentation model trained on languages for which we have large manually annotated corpora. This allows us to apply our approach to less-resourced languages without any manual annotation, with the audio signal as input only. In the present work, the automatic segmentation system was trained for English, language with a lot of resources.

We tested our approach on three languages: American English, French and the less-represented language called Xitsonga. Concerning the results, there are differences according to the target language, and especially according to their characteristics. In all our experiments, the results on the BUCKEYE corpus, which is comprised of conversational speech, are worse than for the other two corpora, which are made up of read speech. The increase in the number of speakers also can be a factor of performance decrease.

With the Xitsonga corpus, for instance, with manual and automatic segmentations, we were able to obtain 46% and 42% purity scores, respectively, at phone-level with 30 pseudo-phones. Based on the inferred pseudo-phones, we discovered about 200 pseudo-words.

Our next work will focus on use DPGMM instead of k-means and on use unsupervised segmentation to have a fully unsupervised model. Furthermore, we presented first results on pseudo-word discovery based on mining similar pseudo-phone sequences. The next step will be to apply pseudo-word discovery algorithms to audio recordings, such as dotplots.

References

1. Towards spoken term discovery at scale with zero resources. In: INTERSPEECH, pp. 1676–1679. International Speech Communication Association (2010)
2. Badino, L., Canevari, C., Fadiga, L., Metta, G.: An auto-encoder based approach to unsupervised learning of subword units. In: ICASSP, pp. 7634–7638 (2014)
3. Badino, L.: Phonetic context embeddings for DNN-HMM phone recognition. In: Interspeech 2016, 17th Annual Conference of the International Speech Communication Association, San Francisco, CA, USA, September 8–12, pp. 405–409 (2016)
4. Church, K.W., Helfman, J.I.: Dotplot: a program for exploring self-similarity in millions of lines of text and code. J. Comput. Graph. Stat. **2**(2), 153–174 (1993)
5. van Heerden, C., Davel, M., Barnard, E.: The semi-automated creation of stratified speech corpora (2013)
6. Kiesling, S., Dilley, L., Raymond, W.D.: The variation in conversation (vic) project: creation of the buckeye corpus of conversational speech. In: Language Variation and Change, pp. 55–97 (2006)
7. Lyzinski, V., Sell, G., Jansen, A.: An evaluation of graph clustering methods for unsupervised term discovery. In: INTERSPEECH, pp. 3209–3213. ISCA (2015)
8. Manenti, C., Pellegrini, T., Pinquier, J.: CNN-based phone segmentation experiments in a less-represented language (regular paper). In: INTERSPEECH, p. 3549. ISCA (2016)

9. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press, Cambridge (2008)
10. Muscariello, A., Bimbot, F., Gravier, G.: Unsupervised Motif acquisition in speech via seeded discovery and template matching combination. IEEE Trans. Audio Speech Lang. Process. **20**(7), 2031–2044 (2012). https://doi.org/10.1109/TASL.2012.2194283
11. Park, A.S., Glass, J.R.: Unsupervised pattern discovery in speech. IEEE Trans. Audio Speech Lang. Process. **16**(1), 186–197 (2008)
12. Pitt, M., Dilley, L., Johnson, K., Kiesling, S., Raymond, W., Hume, E., Fosler-Lussier, E.: Buckeye corpus of conversational speech (2nd release) (2007). www.buckeyecorpus.osu.edu
13. Renshaw, D., Kamper, H., Jansen, A., Goldwater, S.: A comparison of neural network methods for unsupervised representation learning on the zero resource speech challenge. In: INTERSPEECH, pp. 3199–3203 (2015)
14. Tian, F., Gao, B., Cui, Q., Chen, E., Liu, T.Y.: Learning deep representation for graph clustering, pp. 1293–1299 (2014)
15. Versteegh, M., Thiollire, R., Schatz, T., Cao, X.N., Anguera, X., Jansen, A., Dupoux, E.: The zero resource speech challenge 2015. In: INTERSPEECH, pp. 3169–3173 (2015)
16. Vincent, P., Larochelle, H., Bengio, Y., Manzagol, P.A.: Extracting and composing robust features with denoising autoencoders. In: ICML, pp. 1096–1103. ACM (2008)
17. Wang, H., Lee, T., Leung, C.C.: Unsupervised spoken term detection with acoustic segment model. In: Speech Database and Assessments (Oriental COCOSDA), pp. 106–111. IEEE (2011)