Noname manuscript No.
(will be inserted by the editor)

Evolutionary Discriminative Confidence Estimation for Spoken Term Detection

Javier Tejedor · Alejandro Echeverría · Dong Wang · Ravichander Vipperla

Received: date / Accepted: date

Abstract Spoken term detection (STD) is the task of searching for occurrences of spoken terms in audio archives. It relies on robust confidence estimation to make a hit/false alarm (FA) decision. In order to optimize the decision in terms of the STD evaluation metric, the confidence has to be discriminative. Multi-layer perceptrons (MLPs) and support vector machines (SVMs) exhibit good performance in producing discriminative confidence; however they are severely limited by the continuous objective functions, and are therefore less capable of dealing with complex decision tasks. This leads to a substantial performance reduction when measuring detection of out-of-vocabulary (OOV) terms, where the high diversity in term properties usually leads to a complicated decision boundary.

In this paper we present a new discriminative confidence estimation approach based on evolutionary discriminant analysis (EDA). Unlike MLPs and SVMs, EDA uses the classification error as its objective function, resulting in a model optimized towards the evaluation metric. In addition, EDA combines heterogeneous projection functions and classification strategies in decision making, leading to a highly flexible classifier that is capable of dealing with complex decision tasks. Finally, the evolutionary strategy of EDA re-

Javier Tejedor
Human Computer Technology Laboratory, Universidad Autónoma de Madrid
E-mail: javier.tejedor@uam.es

Alejandro Echeverría
Machine Learning Group, Universidad Autónoma de Madrid
E-mail: alejandro.e.rey@gmail.com

Dong Wang
Multimedia Communications Department, EURECOM
E-mail: dong.wang@ed.ac.uk

Ravichander Vipperla
Multimedia Communications Department, EURECOM
E-mail: ravichander.vipperla@eurecom.fr

duces the risk of local minima. We tested the EDA-based confidence with a state-of-the-art phoneme-based STD system on an English meeting domain corpus, which employs a phoneme speech recognition system to produce lattices within which the phoneme sequences corresponding to the enquiry terms are searched. The test corpora comprise 11 hours of speech data recorded with individual head-mounted microphones from 30 meetings carried out at several institutes including ICSI; NIST; ISL; LDC; the Virginia Polytechnic Institute and State University; and the University of Edinburgh. The experimental results demonstrate that EDA considerably outperforms MLPs and SVMs on both classification and confidence measurement in STD, and the advantage is found to be more significant on OOV terms than on in-vocabulary (INV) terms. In terms of classification performance, EDA achieved an equal error rate (EER) of 11% on OOV terms, compared to 34% and 31% with MLPs and SVMs respectively; for INV terms, an EER of 15% was obtained with EDA compared to 17% obtained with MLPs and SVMs. In terms of STD performance for OOV terms, EDA presented a significant relative improvement of 1.4% and 2.5% in terms of average term-weighted value (ATWV) over MLPs and SVMs respectively.

Keywords Spoken term detection · confidence measurement · evolutionary discriminant analysis

1 Introduction

The ever increasing volume of audio data available on the web substantially promotes research on automatic indexing and retrieval of spoken documents. Spoken term detection (STD) is a fundamental task in this direction [35], and was defined by NIST as *searching vast, heterogeneous audio archives for occurrences of spoken terms* [35]. Due to the importance of theoretical research and its potential in practical applications, STD has attracted much interest of late from the likes of IBM [27,26,8]; BBN [19]; SRI & OGI [51,50,1]; BUT [47, 44,45]; Microsoft Research Asia [30]; QUT [49,52]; JHU [36,24,37]; Fraunhofer IAIS/NTNU/TUD [40]; NTU [9,11]; IDIAP [34] etc.

The common STD architecture consists of three main components, as depicted in Fig. 1: a speech recognition component which converts input speech to word or sub-word lattices; a term detector which searches the lattices for potential occurrences of search terms, and a decision maker which evaluates the detected occurrences and hypothesizes reliable ones as output. It is important to note that the speech recognition runs just once on the audio and the term detector does not require the original audio when serving queries.

In STD, a hypothesized occurrence is called a *detection*; if the detection corresponds to an actual occurrence, it is called a *hit*, otherwise it is a *false alarm (FA)*. If an actual occurrence is not detected, this is called a *miss*. To evaluate the STD performance, NIST defines a metric called *average term-weighted value (ATWV)* [35] and a *detection error tradeoff (DET) curve* [29]

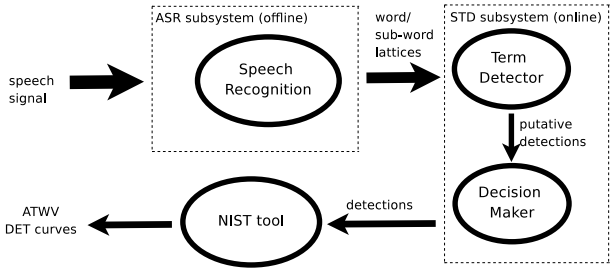

Fig. 1 *The standard STD architecture: a speech recognizer converts speech into word/sub-word lattices; a term detector searches for potential occurrences of search terms; a decision maker determines whether a detection is reliable. The NIST tool is used to evaluate detection performance.*

which works at various hit/FA ratios. Both ATWV and DET curves are used for performance evaluation in this paper.

Within the STD architecture, the decision maker plays an important role in determining eligible detections, which is usually based on certain confidence measures. Term-dependent confidence measures derived from discriminative models, such as a multi-layer perceptron (MLP) or a support vector machine (SVM), have been shown to outperform the commonly used lattice-based confidence [56]. Generally speaking, this discriminative approach treats the hit/FA decision as a two-class classification problem, and derives confidence measures from classification posterior probabilities. We will show that discriminative confidence is an essential requirement for ATWV-oriented decision making and is inherently consistent with the confidence normalization technique proposed in [56].

Discriminative confidence can be derived from any discriminative model, though MLPs and SVMs are the most commonly used. A possible drawback of MLPs and SVMs, however, is that their cost functions are based on some intermediate metrics instead of relying on the classification error rate itself. For example, MLPs take cross entropy as their objective function, while SVMs maximize the minimum soft margin of training patterns to the decision boundary. More importantly, these objective functions are all continuous, which greatly limits the discriminative power of these models in complex tasks where the decision boundary is highly complicated. Another problem, mainly for MLPs, is that the training process depends heavily on initialization, and is therefore more likely to be trapped in a local minimum. These disadvantages result in considerable performance degradation when measuring detections of the out-of-vocabulary (OOV) terms, for which the term properties (pronunciation variation, occurrence rate, confidence distribution, ASR error pattern) tend to be more diverse compared to the in-vocabulary (INV) terms [55,57], and therefore the decision boundary tends to be more complicated.

We propose a new discriminative confidence estimation approach based on an evolutionary algorithm, named evolutionary discriminant analysis (EDA). Unlike MLPs and SVMs, EDA uses the classification error rate as its objective,

which on one hand removes the continuity assumption on objective functions of MLPs and SVMs, and on the other hand optimizes the evaluation metric directly. Moreover, EDA combines heterogeneous projection functions and classification strategies in decision making, which empowers EDA to handle very complex decision boundaries. Finally, the intrinsic randomness within the evolution approach provides a simple mechanism to rescue models trapped in local minima. We argue that these advantages make EDA a better model for discriminative confidence estimation for STD than standard MLPs and SVMs, especially for OOV terms for which the decision boundary is complex.

The authors note that EDA has previously been applied to a number of other applications, and substantial success has been obtained in solving problems with highly mixed features that are unevenly distributed with multiple modes (e.g., classification on complex data such as UCI databases [41,42]). The novelty of this paper from the EDA perspective is that EDA is extended to provide classification posterior probabilities instead of making hard classification decisions. To the best of our knowledge, this is the first effort to apply evolutionary approaches to STD.

A work related to EDA is the evolutionary MLP training [18,38,39], in which MLP parameters are learnt in an evolutionary manner. The difference between EDA and this approach is that EDA minimizes the classification errors and hence optimizes the task objective directly, while evolutionary MLP training minimizes the mean square error, which is an intermediate objective and is thus closer in principle to the conventional MLPs. Readers are encouraged to refer to [41,42] for more details distinguishing these two approaches.

The rest of the paper is organized as follows: we first describe the discriminative confidence estimation in STD in Section 2, and then present the evolutionary algorithm in Section 3. In Section 4 we present the implementation of the EDA approach for the discriminative confidence estimation, we report our experiments in Section 5, and the work is concluded in Section 6 with some ideas for future work.

2 Discriminative confidence estimation in spoken term detection

As shown in Fig. 1, the decision maker plays an important role in STD: it determines if a detection is reliable or not. This is named as a *hit/FA decision*. In most cases, this decision is based on some form of confidence measure, or simply *confidences*. To make the presentation clear, we denote a detection d as a tuple which captures all the available information:

$$d = (K, s = (t_s, t_e), v_a, v_l, ...) \tag{1}$$

where v_a, v_l represent the acoustic likelihood and language model score respectively and s denotes the speech segment from t_s to t_e where the detection of the term K resides. Other informative factors, such as term occurrence rates, are represented by "...". The task of the confidence estimation based on this representation then amounts to deriving a certain confidence measure from the

information encapsulated in the tuple d. A widely used confidence measure is the detection posterior probability, which was proposed by Wessel et al. [59] and found widespread use in STD research [32, 27, 46, 50, 30]. It can be formally written as follows:

$$c_{lat}(d) = P(K_{t_s}^{t_e}|O) \tag{2}$$

where $K_{t_s}^{t_e}$ denotes the event of term K appearing in the speech segment from t_s to t_e, and O represents the audio stream. In practice, this confidence is often approximated by the lattice posterior probability, and hence is also called the *lattice-based confidence* [59]. We use c_{lat} to denote the lattice-based confidence measure in Eq. 2.

2.1 OOV challenge and confidence normalization

Although the lattice-based confidence performs well in many STD tasks, e.g., [27, 46, 50], it exhibits severe performance reduction when measuring detection of OOV terms, as shown in [56]. This motivates a thorough study on OOV terms.

In STD, OOV words are those words absent from the system dictionary, and OOV terms are those containing one or more OOV words. INV terms, correspondingly, are those terms containing only in-vocabulary words. Some words are OOV simply because the system vocabulary has a fixed size, whereas others arise from the dynamics of human language evolution. One estimate is that about 20,000 new words are coined each year [58]. OOV terms present a significant challenge to STD; in one real spoken document retrieval system, 12% of queries were reported to contain OOV terms [25]. Since new words are continually being created, an STD system, even with a very large vocabulary, will eventually receive a significant number of OOV queries.

A widely adopted approach to OOV STD is based on sub-word units, usually phonemes [47, 27, 1]. In this approach, phoneme transcriptions of OOV terms are searched for in the phoneme lattices generated by a phoneme-based speech recognizer. Unlike the INV terms for which the phone transcriptions can be obtained from the system dictionary, OOV terms have to resort to letter-to-sound (LTS) conversion. State-of-the-art LTS for English presents a word error rate in the order of 30% [15, 6, 14, 48, 4]. This means that OOV STD is actually based on uncertain phoneme transcriptions, which in turn leads to more noisy detections and more complex error patterns.

Another challenge from OOV terms relates to the high diversity in term properties. Adopted from different sources, OOV terms usually possess highly varying properties in various aspects, e.g., occurrence rate, phonemic structure, linguistic background, morphological form, etc. This diversity is more evident for open languages such as English. This diversity in term properties certainly results in diverse patterns in confidence measures, which in turn lead to complex decision boundaries in hit/FA decision making. To illustrate the

term diversity and in particular for the OOV terms, the term occurrence distribution for both the OOV and INV terms are shown in Fig. 2 and Fig. 3 respectively. As expected, we can see a larger variation in the distribution of occurrences of OOV terms as compared to that of INV terms: while the INV terms have occurrences varying from 5 to 20, the OOV terms have occurrences varying from 1 to more than 300, with less frequent terms in domination.

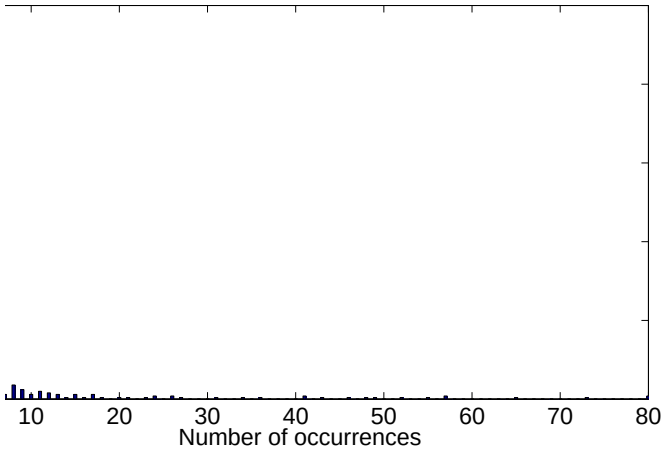

Fig. 2 *Histogram of occurrences of the* 484 *OOV terms. The X-axis shows the number of occurrences and the Y-axis shows how many terms have each number of occurrences in the evaluation set. The X-axis corresponding to* 80 *refers to all the terms with a number of occurrences equal to or greater than* 80*.*

Another example of OOV term diversity is related to phonetic regulation. Some OOV terms follow the English spelling and pronunciation rules well, e.g., 'GOOGLE', while others are simply out-of-rule, particularly those borrowed from other languages, e.g., 'OKINAWA'. Regular OOV terms can easily derive their pronunciations by LTS and their phoneme fragments tend to appear in the training data and therefore are partially represented by the acoustic and language models; in contrast, irregular OOV terms usually obtain unreliable pronunciations and their phoneme fragments are often missed in the training data. We show that this phonetic regulation diversity leads to different patterns for INV and OOV terms with differences in the n-gram occurrences for language model training and in the distribution of the triphone occurrences for acoustic model training in Fig. 4 and Fig. 5 respectively. From Fig. 4, it can be seen that there are fewer occurrences of OOV n-grams than INV n-grams in the training data particularly when n is high. From Fig. 5, it can be seen that fewer training instances are available for triphones of OOV terms than those of INV terms, and the distributions are slightly different. The OOV terms display a flatter distribution, indicating higher diversity among OOV terms.

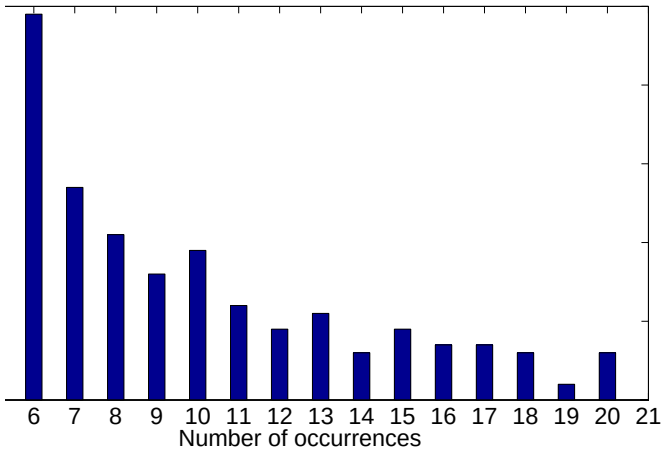

Fig. 3 *Histogram of occurrences of the* 256 *INV terms. The X-axis shows the number of occurrences and the Y-axis shows how many terms have each number of occurrences in the evaluation set.*

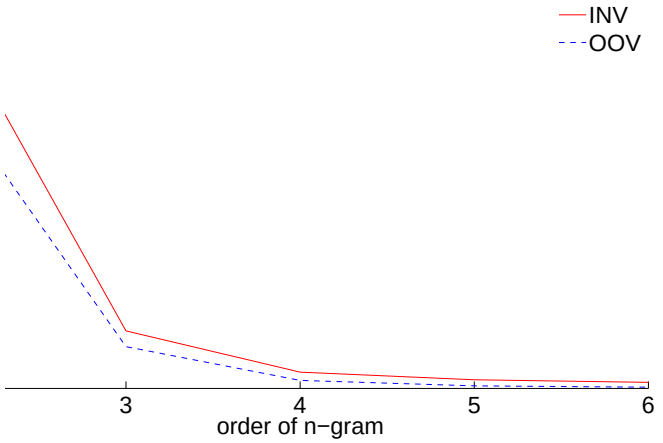

Fig. 4 *The average number of occurrences of various orders of n-grams in the training corpus. INV denotes in-vocabulary terms and OOV denotes out-of-vocabulary terms.*

A possible way to deal with this OOV challenge is to consider the term identity in decision making, which is the idea of the term-specific threshold (TST) approach [32]. An alternative way is to normalize the term-independent lattice-based confidence with term-dependent filters so that the term diversity can be compensated for. This approach is known as *confidence normalization* [56]. Due to its importance in developing the discriminative confidence estimation, we outline the normalization technique in the following.

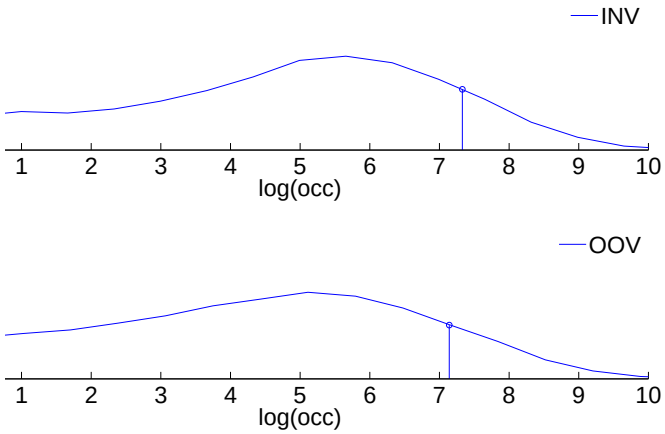

Fig. 5 *The distributions of the occurrences (in log scale) of INV triphones (top plot) and OOV triphones (bottom plot). The vertical stems represent average occurrences.*

We start with the definition of the ATWV metric, which integrates the miss rate and false alarm rate of each term into a single metric and then averages over all the search terms:

$$ATWV = \frac{1}{|\Delta|} \sum_{K \in \Delta} \left(\frac{N_{hit}^K}{N_{true}^K} - \beta \frac{N_{FA}^K}{T - N_{true}^K} \right) \tag{3}$$

where Δ denotes the set of search terms and $|\Delta|$, the number of terms in this set. N_{hit}^K and N_{FA}^K represent the number of hits and false alarms of the term K respectively and N_{true}^K is the number of actual occurrences of K in the audio. T denotes the audio length in seconds, and β is a weight factor.

It can be rearranged as follows:

$$ATWV = \frac{1}{|\Delta|} \sum_{K \in \Delta} \sum_{i} \left(\frac{I(d_i^K)}{N_{true}^K} - \beta \frac{1 - I(d_i^K)}{T - N_{true}^K} \right) \tag{4}$$

where d_i^K denotes the i-th detection of the term K, and $I(d)$ is an indicator function which takes 1 if d is a hit and 0 otherwise. Taking expectation on both sides of the equation, and noticing that the expected occurrence of a detection in a certain class corresponds to the posterior probability that the detection belongs to that class, we have

$$E(ATWV) = \frac{1}{|\Delta|} \sum_{K \in \Delta} \sum_{i} \left(\frac{P(C_{hit}|d_i^K)}{N_{true}^K} - \beta \frac{1 - P(C_{hit}|d_i^K)}{T - N_{true}^K} \right) \tag{5}$$

where $P(C_{hit}|d_i^K)$ represents the posterior probability that d_i^K belongs to the class of hits (C_{hit}). For any detection d_i^K, the quantity in the parentheses

corresponds to its contribution to ATWV if it is treated as a hit. According to decision theory, d_i^K should be classified as a hit if the contribution is positive, i.e.,

$$\frac{P(C_{hit}|d_i^K)}{N_{true}^K} - \beta\frac{1-P(C_{hit}|d_i^K)}{T-N_{true}^K} > 0. \tag{6}$$

A simple rearrangement leads to

$$\xi(P(C_{hit}|d_i^K)) = P(C_{hit}|d_i^K) - \frac{\beta N_{true}^K}{(\beta-1)N_{true}^K + T} > 0 \tag{7}$$

where the function ξ represents confidence normalization since it compensates for the term-dependent occurrences N_{true}^K. This normalization, as shown in [56], is very effective in dealing with the high diversity among OOV terms. Note that N_{true}^K is unknown in practice and thereby has to be estimated from the data. As in the TST approach, we use the expected count as the estimate:

$$N_{true}^K \approx \sum_i P(C_{hit}|d_i^K). \tag{8}$$

2.2 Discriminative confidence

The derivation of the confidence normalization technique in the previous section indicates that an optimal decision in terms of ATWV should be based on the normalized classification posterior probability $\xi(P(C_{hit}|d_i^K))$, henceforth denoted as $\xi(P(C_{hit}|d))$. From the perspective of the confidence estimation, $P(C_{hit}|d)$ is named as a discriminative confidence since it is discriminative for hits and FAs. We denote the discriminative confidence by c_{disc}, formally written as follows:

$$c_{disc}(d) = P(C_{hit}|d).$$

Accordingly, the normalized posterior probability is named as the normalized discriminative confidence, given by

$$\hat{c}_{disc}(d) = \xi(P(C_{hit}|d))$$

where ξ is as defined in Eq. 7.

The discriminative confidence estimation was first presented in [56], where MLPs and SVMs are employed to derive the posterior probabilities. Following the same notation, the approach can be formally represented as a non-linear mapping f from a set of informative features to $c_{disc}(d)$:

$$f : (c_{lat}(d), A, L, T, R_0(K), R_1(K)) \longrightarrow c_{disc}(d) \tag{9}$$

where $c_{lat}(d)$ is the lattice-based confidence. The rest of the input features include the acoustic likelihood (A), the language model score (L), the duration

of the detection (T), and two term-dependent features $R_0(K)$ and $R_1(K)$ defined as follows:

$$R_0(K) = \frac{\sum_i c_{lat}(d_i^K)}{T_0} \tag{10}$$

and

$$R_1(K) = \frac{\sum_i (1 - c_{lat}(d_i^K))}{T_0} \tag{11}$$

where T_0 is the length of the audio. Note that R_0 and R_1 are designed to introduce term-dependency (occurrence rates here) in the modeling, and are motivated by the definition of ATWV.

A particular advantage of this approach is that term-dependent factors (such as the R_0 and R_1 in Eq. 9) can be involved in the modeling and hence are taken into account in confidence measuring. This is a more flexible way to compensate for term-dependent factors than the normalization technique where only the term occurrences are taken into account. Therefore it is not surprising that the discriminative confidence provides a considerable performance improvement particularly for OOV terms, as reported in [56].

3 Evolutionary Algorithms and Evolutionary Discriminant Analysis

A potential problem of MLPs and SVMs is that they take some intermediate measurements as objective functions, instead of the evaluation metric, i.e., classification error rate (CER). For MLPs, the objective is maximum cross entropy while for SVMs the objective is maximum soft margin. These intermediate objectives could possibly lead to a sub-optimal model training in terms of the evaluation metric; more importantly, they all impose some continuity assumptions: for MLPs, the posterior probability is assumed to be continuous and for SVMs the slack penalty is assumed to be continuous. This artificial assumption may lead to considerable performance degradation in tasks where the decision boundary is complex. For instance, in OOV STD where the hit/FA decision making is complicated, the continuity assumption seems to be over strict. A better model, of course, should take the CER as its objective *directly*. Such a model consists of at least two components: first a projection component maps patterns to a projection space, and then a classification component classifies the mapped patterns according to a certain classification strategy. This is illustrated in Fig. 6.

The CER-oriented optimization for this model can be formulated as follows:

$$\hat{\theta} = arg \min_{\theta} \sum_{d} \delta\{H_{g_\theta}(d), t(d)\} \tag{12}$$

where d is a training pattern, and $t(d)$ is its class label; g_θ is a projection function depending on some parameters θ, H_{g_θ} is a classification function in

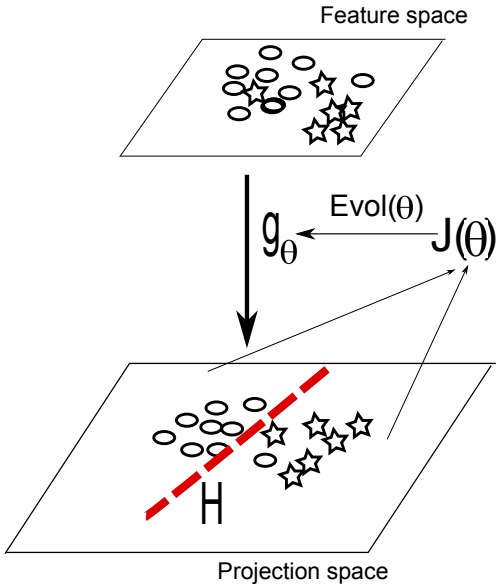

Fig. 6 *Illustration of the EDA approach. The small circles and stars represent training patterns of two classes. The dashed line in the projection space represents the decision boundary in the projection space according to the classification function H. $Evol(\cdot)$ denotes an evolutionary strategy.*

the projected space, and $\delta(a, b)$ is an indicator function which is equal to 0 if $a = b$ and equal to 1 otherwise.

The main obstacle with this model, however, is that the objective function will never be continuous, and so the conventional gradient or quadratic approaches for the model training do not work. A possible solution is to exploit the evolution strategy [3] to 'breed' some solutions and then choose the optimal one.

3.1 Evolutionary algorithm

An evolutionary algorithm (EA) is based on a number of optimization techniques that resemble the mechanisms of biological evolution: reproduction, mutation, and selection. With this approach, a candidate solution corresponds to an individual *chromosome*, and the goodness of the solution corresponds to the fitness of the chromosome and is determined by a fitness function. In our case, we choose the classification error as the fitness function since it is directly related to the evaluation metric of our task (i.e., CER and ATWV). Other fitness functions are possible, e.g., square distance; the appropriate choice is fully task-dependent and has a direct impact on the model quality and the convergence speed. An evolutionary computing process starts from a set of random initial solutions, resembling the starting population of a biological system; parents with high fitness are then chosen to beget the offspring by copying

and crossover recombination of their genes, alike biological reproduction. This reproduction involves a random modification of the genes copied from their parents, similar to mutation in biological evolution. The new generation competes for survival in a limited population according to the fitness. This process is repeated until a candidate with sufficient quality is found or a previously defined computational limit is reached, which results in an optimal solution.

Evolutionary algorithms often perform well in all types of problems because they typically do not make any assumption about the underlying fitness landscape; this generality has led to success in broad fields such as engineering [31,28], biology [21], economics [12], physics [2], medicine [43], ecology [33], information retrieval [13], etc.

3.2 Evolutionary Discriminant Analysis

Instead of resorting to gradient-based optimization or quadratic programming as with MLPs and SVMs, evolutionary algorithms optimize objective functions by selecting, recombining and mutating existing solutions according to the fitness function which is not necessarily continuous, and therefore can be used to solve the optimization problem defined by Eq. 12. One approach following this strategy has been proposed by Sierra et al. in the name of evolutionary discriminant analysis (EDA) [41,42]. As shown in Fig. 6, this approach involves a projection function g_θ which is continuous and non-linear and is parametrized by θ. An instance of θ is then regarded as a chromosome. The training process starts from projecting all the training patterns to a projection space by g_θ, where the projected patterns are classified according to the classification function H. The fitness of the chromosome θ, denoted by $J(\theta)$, is measured by the classification error rate. In order to find a θ which optimizes the fitness, a number of instances of θ are randomly initialized, forming an initial population. New instances of θ are then reproduced based on the existing population by recombining a few randomly selected parents plus mutation noise. Only the most promising offspring are retained, just like the natural selection process in biological systems. This evolutionary process continues until no fitness improvement is observed within a prescribed number of batch generations[1]; therein the best instance of θ represents the optimal solution of the parameters of the EDA model.

EDA resembles some linear discriminants such as the Fisher discriminant [20] in the sense that the input patterns are first projected to a lower dimension space and then classified by some conventional classification approach. However EDA takes the CER as its objective whilst Fisher discriminant seeks maximum data separation. More importantly, EDA optimizes the projection

[1] This criterion is widely used in Evolutionary Computing to test the algorithm convergence [17]. The underlying idea is that if evolution does not improve the fitness over a large number of generations, then there is a high probability that an optimal solution has been attained.

by taking classification into consideration, which actually forms an integrative optimization. The Fisher discriminant, in contrast, merely optimizes the projection. Finally, the Fisher discriminant assumes a linear projection, whilst EDA can choose any form of projection function, leading to much more flexibility.

Compared to other non-linear classifiers such as MLPs and SVMs, EDA removes the underlying continuity assumption by allowing discrete objectives, which allows it to solve tasks with non-differentiable complex decision boundaries; moreover, EDA uses the evaluation metric (CER) as its objective, which usually results in a higher performance in terms of this metric. Finally, the randomness inherent in EDA reduces the risk of local minima, avoiding a critical problem that impacts many non-linear classifiers such as an MLP.

The advantage of EDA is largely attributed to the flexible combination of the continuous projection function and the discrete classification function in its model structure, which lends EDA the capability to deal with highly complex decision boundaries. The evolutionary approach ensures that this complex objective can be optimized, although at the cost of increased computational requirement.

4 EDA for confidence estimation

In this section, we apply EDA to discriminative confidence estimation for STD. We first present how to construct an EDA solution for a classification task based on an MLP-style non-linear projection and a nearest-neighbor classifier. The evolution procedure is then outlined, and the hard decision making (classification) is extended to posterior probability estimation.

4.1 Evolutionary treatment

In order to construct an evolutionary solution for a classification task with EDA, we need to specify the projection function g and the classification function H. The instances of the parameters θ of the projection function are then treated as chromosomes and are optimized with the evolution program.

4.1.1 MLP-based projection

The EDA approach allows an arbitrary projection function, though a non-linear projection is preferred due to its capacity to fit complex decision boundaries. In our implementation, an MLP-style non-linear function is used due to its ability to approximate any continuous function with compact parameters.

We choose a 3-layer MLP structure as shown in Fig. 7, which consists of $N+1$ input units, $M+1$ hidden units and K output units, where the notation $+1$ indicates the bias unit in the input and the hidden layers. The weights of the first layer are denoted by w_{nm}, where $n = 0, \ldots, N$ and $m = 1, \ldots, M$,

and the weights of the second layer are denoted by v_{mk}, where $m = 0, \ldots, M$ and $k = 1, \ldots, K$. The active function applied to the hidden units is a logistic sigmoid, i.e.,

$$\varphi(z) = 1/(1 + \exp^{-z}) \tag{13}$$

and the active function applied to the output units is linear.

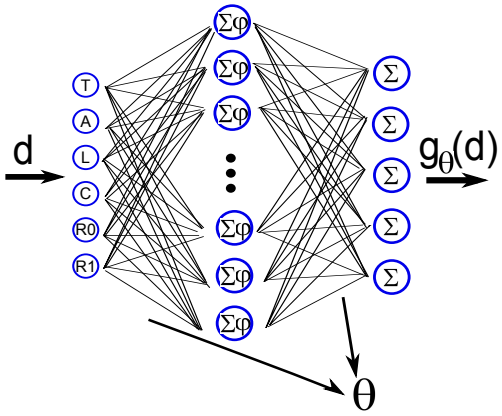

Fig. 7 *The MLP-style non-linear projection function. The input consists of* 6 *features and the projected space has* 5 *dimensions. The logistic sigmoid activation has been applied to the hidden units.*

With this MLP-alike non-linear projection, a pattern d in the N-dimensional feature space is projected to $\hat{d}$ in the K-dimensional projection space, where the k-th dimension $\hat{d}(k)$ is represented by the k-th output of the MLP structure. This is formulated as:

$$\hat{d}(k) = \sum_{m=0}^{M} v_{mk} \varphi(\sum_{n=0}^{N} d(n) w_{nm}) \tag{14}$$

$$= g_\theta^k(d) \tag{15}$$

where $d(n)$ represents the n-th feature of d including the bias, φ is the logistic sigmoid function defined in Eq. 13 and g_θ^k represents the projection function on the k-th dimension of the projection space. The entire set of projection functions is a collection of all these single-dimension projections, i.e., $g_\theta = \{g_\theta^k\}$, where $\theta = \{w_{nm}\} \cup \{v_{mk}\}$. A candidate solution of θ corresponds to a chromosome in EDA, and the optimal solution needs to be discovered by the evolution strategy that will be presented in Section 4.2.

It should be emphasized that the MLP-alike structure here is used simply to represent a non-linear function; it is not a real MLP. Firstly, the output does not associate any regression or classification targets – actually, the number of output units is the dimension of the projected space and is not equal to the

number of classes in general. And secondly, the update of the weights is driven by evolution, instead of by error back-propagation [5].

4.1.2 Classification and fitness

The second component of the EDA approach is the classification function H, which directly relates to the fitness of chromosomes/solutions. We choose the nearest neighbor classifier in this study, which assigns a pattern to the class whose mean is nearest to the pattern in the projection space.

Specifically, the mean of each class is computed in the N-dimensional feature space and is then mapped to the K-dimensional projection space:

$$\hat{m}_r = g_\theta(\frac{1}{|\Re_r|}\sum_{d_i \in \Re_r} d_i) \qquad r = 1, \ldots, R \tag{16}$$

where $\Re_r$ denotes the set of training patterns for the r-th class. In our STD task, the patterns belong either to hits or false alarms, and therefore R=2. The reason we compute class means in the input space and then project them to the projection space (using Eq. 16) instead of computing them in the projection space directly is that this enables a much faster EDA, as we can obtain the projection means by a simple non-linear function calculation instead of costly data pooling each time the projection is updated.
Applying the same projection to each training pattern d, we have its image $\hat{d}$ in the projection space, formally represented as:

$$\hat{d} = g_\theta(d). \tag{17}$$

The nearest-neighbor approach is then employed to classify a pattern by assigning it to a class whose mean projection is nearest to the projection of the pattern. This results in the classification function H as follows:

$$H_{g_\theta}(d) = \arg\min_r ||\hat{d} - \hat{m}_r||_2^2 \tag{18}$$

where $||\cdot||_2$ is the Frobenius norm. Although other competing classification approaches could have been exploited, we have chosen the nearest neighbor classifier due to its simplicity and computational efficiency. With the projection flexible enough, we find that this simple classifier provides fairly good performance.
The classification errors, or the fitness of the chromosome θ, is then given by

$$J(\theta) = \sum_d \delta(t(d), H_{g_\theta}(d)) \tag{19}$$

where δ is the indicator function as in Eq. 12.

In order to reduce the risk of over-fitting, we partition the training data into a training set and a validation set. The class means are simply computed based on the training set, while the fitness is computed on both the training set and the validation set, leading to a composite fitness function:

$$J_{all}(\theta) = J_{tr}(\theta) + J_{va}(\theta) \tag{20}$$

where $J_{tr}(\theta)$ and $J_{va}(\theta)$ are the classification errors computed according to Eq. 19 on the training set and the validation set respectively.

4.2 Evolutionary program

We have cast a classification task to an EDA problem in the previous section; now the evolutionary strategy can be employed to search for the optimal chromosome, or the classification parameters θ. Algorithm 1 illustrates the evolutionary process, where $\Im$ represents a population, and $\mu, \rho, \lambda, \sigma, \aleph$ are the parameters controlling the evolution process. The algorithm starts by randomly sampling μ chromosomes as the initial population, and then evolves the population by reproducing new offspring and selecting the most promising as the next generation. This reproduction-selection process continues until the convergence criterion is satisfied. $Reproduce(\Im_{old}, \lambda, \sigma, \rho)$ is the function that conducts reproduction, and $Select(\Im, \mu)$ is the function that performs selection, i.e., selects μ chromosomes from $\Im$ according to their fitness (Eq. 20). $Fitness(\Im)$ is an auxiliary function that returns the best fitness of the chromosomes in the population $\Im$.

Algorithm 1 EDA algorithm

Require: $\mu, \rho, \lambda, \sigma, \aleph$
{μ: population size}
{ρ: size of family, i.e., number of parents to reproduce an offspring}
{λ: total number of offspring in reproduction}
{σ: mutation noise}
{$\aleph$: number of batch generations}
$\Im_{old}$=Init(μ)
J_{best}=MAX_FLOAT
i=**0**;
while $\aleph > i$ **do**
  $\Im_{new}$=Reproduce($\Im_{old}$,ρ,λ,σ)
  $\Im_{old}$=Select($\Im_{new}$,μ)
  J_{new}=Fitness($\Im_{old}$)
  if $J_{new} < J_{best}$ **then**
    i=**0**;
    $J_{best} \leftarrow J_{new}$
  else
    $i \leftarrow i + 1$
  end if
end while

The production process (function *Reproduce*) is presented in Algorithm 2. An offspring θ is produced by recombining its ρ parents that are randomly selected from the current generation $\Im$. Each gene in θ is copied from one of its parents plus a mutation noise ι sampled from the Gaussian distribution

Algorithm 2 Reproduction in EDA

Require: $\Im, \rho, \lambda, \sigma$
1: {$\Im$: the existing population of size μ}
2: {ρ: size of family, i.e., number of parents to reproduce an offspring}
3: {λ: total number of offspring in reproduction}
4: {σ: mutation noise}
5: $\Im_{new}$={}
6: **for** i:=1 to λ **do**
7: Ξ=sampling($\Im$,ρ)
8: θ=**0**;
9: **for** j:=1 to length(θ) **do**
10: ξ=sampling(Ξ)
11: $\theta_j = \xi_j + \iota \quad \iota \sim N(0, \sigma)$
12: **end for**
13: $\Im_{new} \leftarrow \Im_{new} + \theta$
14: **end for**
return $\Im_{new}$

$N(0, \sigma)$. The production process stops after λ offspring are produced. Thereafter a new generation is created and returned, participating in competition for survival in the selection process.

4.3 Discriminative confidence estimation

The EDA model optimized with the evolutionary approach presented in the previous section can be applied directly to classification tasks according to the classification function H (Eq. 18). However in STD, we prefer a 'soft decision' based on discriminative confidence so that the ATWV-oriented decision can be conducted with the normalization technique as presented in Section 2. Therefore, the EDA approach needs to be extended to predict classification posterior probabilities instead of class categories. This can be achieved by measuring the relative distance of the projected detection $\hat{d}$ to the projected means of the hit and FA classes, i.e., $\hat{m}_{hit}$ and $\hat{m}_{FA}$ respectively. As shown in Fig. 8, we first draw a vector from the mean of FAs to the mean of hits, and then project the detection image $\hat{d}$ on to the vector, obtaining the new image $\ddot{d}$.

The posterior probability of d belonging to the hit class, or the discriminative confidence of d, is then given by the following equation.

$$c_{disc}(d) = \begin{cases} 0 & \alpha < 0 \\ \alpha & 0 \leq \alpha \leq 1 \\ 1 & \alpha > 1 \end{cases} \tag{21}$$

where α is represented by the following equation.

$$\alpha = \frac{\ddot{d} - \hat{m}_{FA}}{\hat{m}_{hit} - \hat{m}_{FA}} \tag{22}$$

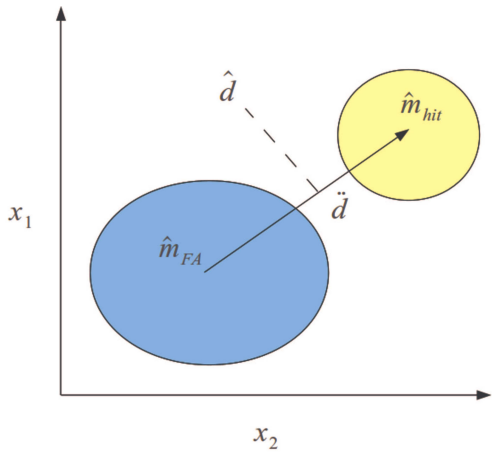

Fig. 8 *EDA-based discriminative confidence estimation. The two shaded circles represent the class of hits and FAs in the projection space, and* $\hat{m}_{hit}$ *and* $\hat{m}_{FA}$ *are the projections of the means of the two classes, hit and FA, respectively.* $\hat{d}$ *represents the projection of an evaluation pattern, and* $\ddot{d}$ *is its image on the vector from* $\hat{m}_{FA}$ *to* $\hat{m}_{hit}$*. The posterior probability of* d *belonging to each class is then proportional to the distance from* $\ddot{d}$ *to the mean projection of that class.*

5 Experiments

The proposed EDA-based discriminative confidence estimation is evaluated on an STD task on an English meeting domain corpus. We first manually selected 256 terms that are entity names with sufficient occurrences from the dictionary used by the AMI LVCSR system [22] as INV terms. These terms, which have 2329 occurrences in the evaluation data, appear in the system dictionaries for speech recognition and term detection and are well represented by the language and acoustic models. OOV terms are strictly defined as those terms absent in the dictionaries of both the ASR and the term detection systems, and absent in training material for acoustic and language models. This means our OOV terms are not only 'out-of-vocabulary', but also 'out-of-language'. To comply with this definition, all occurrences of these terms are purged from the dictionaries as well as from the speech and text corpora used for model training. We first compared the AMI dictionary (in active use and assumed to represent current usage) and the COMLEX Syntax dictionary v3.1 (published by LDC in 1996 and therefore historical from an STD perspective), and selected 414 terms as OOV terms from the AMI dictionary that do not occur in the COMLEX dictionary. These terms were chosen to simulate the evolution of English over time and are referred to as *real* OOV terms since they are created in recent years and are absolutely OOV for any previously developed system. Additionally, in order to design a reliable experiment with sufficient OOV occurrences, we manually selected another 70 *artificial* OOV terms that are plausible as search terms such as city and person names. These terms are not really new in English, but still display OOV properties from the

perspective of our system design, since all their occurrences in the dictionaries and the training material have been removed and their pronunciations have been obtained using LTS conversion. In total, we have 484 OOV terms with 2736 occurrences in the evaluation data.

The speech data used for acoustic model (AM) training, system development and performance evaluation come from multi-party meetings recorded at several institutes, including ICSI; NIST; ISL; LDC; the Virginia Polytechnic Institute and State University; and various partners of the AMI project consortium. The speech data recorded by individual head-mounted microphones (IHM) were used. The original AM training corpus comprises 73 hours of speech from 30 meetings at ICSI [23], 13 hours of speech from 15 meetings at NIST, 10 hours of speech from 19 meetings at ISL [7] and 16 hours of speech from 35 meetings from AMI partners [22]. In total, there are 104 hours of speech with the regions of silence excluded. Next, we purge the OOV terms from them by removing those sentences that contain any OOV term, which removed 23% of the speech data and resulted in our final AM training collection that consists of 122744 utterances with a total duration of about 80.2 hours of speech. The official RT04s development set provided by NIST, which consists of 1.40 hours of speech excerpted from 8 meetings recorded at ICSI, NIST, ISL and LDC was used for parameter tuning. The evaluation set consists of the official RT04s eval set which consists of 1.7 hours of speech from 8 meetings recorded at ICSI, NIST, ISL and LDC, the official RT05s eval set which comprises 2.1 hours of speech recorded from 10 meetings at ICSI, ISL, VT and AMI partners and a speech corpus AMI08 which consists of 7.2 hours of speech from 12 meetings recorded at the University of Edinburgh in the AMIDA project[2]. The total size of the evaluation set is 11 hours of speech. There is no overlapping between training, development and test sets.

The text corpus used to train the language model was provided by the AMI project and is the one used by the AMI RT05s LVCSR system [22]. This corpus contains text from various sources such as news, transcripts of speech corpora and a large amount of web text, amounting to 521.4M words after OOV purging. The 50k AMI dictionary (from which we had purged the OOVs) was used to convert the word-based text corpus to a phoneme-based corpus.

We built a phoneme-based system using the speech and text corpora described above. The acoustic models are state-clustered triphone HMMs with Mel frequency cepstral coefficient (MFCC) features. The language model is a phoneme 6-gram model where the order is chosen as the value that provides the best STD performance on the development set among various models allowed by computational resources [53]. Cambridge University's HTK is used to train the acoustic models and for lattice generation, and the SRI LM toolkit is used to train the LM. An enhanced joint-multigram model [16, 54] trained with the AMI dictionary is used to predict pronunciations for the OOV terms. The *Lattice2Multigram* tool from *Speech@FIT* (Brno University of Technology) is

[2] *http://www.amiproject.org/*

used to search for detections within the phoneme lattices. More information about the experimental setting can be found in [53].

We conduct a comparative study of EDA with the other two discriminative models, an MLP and an SVM. We first describe the data and configurations used to train these models, and then present the results on the classification and STD tasks.

5.1 Model training

In order to train a discriminative model, we need a set of positive and a set of negative training samples, which correspond to hits and false alarms respectively in STD. Therefore the first step in our experiments is to conduct STD on the development set without discriminative confidence and normalization applied. The output detections are then collected together with the informative attributes enumerated in Eq. 9, including the lattice-based confidence, the acoustic likelihood, the language model score, the time duration, and two occurrence-derived attributes $R_0(K)$ and $R_1(K)$. These detections are labeled as hits and false alarms according to the reference, and are used as positive and negative samples to train the MLP, SVM and the EDA. A particular problem in the training set is that there are far more negative samples than positive samples, which results in biased models preferring FAs. To address this imbalance, we duplicate some hits to make them similar in number to FAs, and train balanced models with the balanced data. A standard K-fold cross-validation with $K = 10$ is employed for all the three models.

We choose a 3-layer MLP in this work. The input layer consists of 6 units, corresponding to the 6 informative attributes. The hidden layer consists of 30 units (chosen by cross validation) with sigmoid activation. The output layer consists of two units with soft-max activation, corresponding to hits and FAs respectively. The standard error back-propagation algorithm [5] is employed to train the model. The SVM is trained with the LIBSVM toolkit [10] with a radial basis kernel function. The parameters, including the error penalty C for classification and the radius scale γ for the kernel, are again optimized by cross-validation, giving $C = 32$ and $\gamma = 0.5$ in our experiments.

The EDA training is a bit more complicated. First, the MLP-style projection function needs to be specified. The input layer is fixed and consists of 6 units corresponding to the 6 informative attributes; the hidden layer and the output layer, however, need to be optimized with respect to the fitness value. The cross-validation shows that the optimal structure is composed of 12 hidden units and 5 output units. This means that the optimal projection space for the classification task has 5 dimensions. The parameters that control the evolution process are chosen heuristically, as $\mu = 15$, $\lambda = 100$, $\rho = 2$, $\sigma = 0.15$, $\aleph = 100$.

5.2 Classification

In this experiment, we investigate the performance of various discriminative models on the classification task. The detections obtained from STD conducted on the evaluation set are used to evaluate the three discriminative models. The results are measured in terms of the hit misclassification rate, $\varepsilon(Hit)$, and the FA misclassification rate, $\varepsilon(FA)$, defined as follows:

$$\varepsilon(Hit) = 1 - \frac{\sum_K \hat{N}_{hit}^K}{\sum_K N_{hit}^K}$$

and

$$\varepsilon(FA) = 1 - \frac{\sum_K \hat{N}_{FA}^K}{\sum_K N_{FA}^K}$$

where N_{hit}^K and N_{FA}^K are the number of hits and FAs of the term K in the evaluation data according to the reference, and $\hat{N}_{hit}^K$ and $\hat{N}_{FA}^K$ are the number of hits and FAs of the term K that are correctly classified by the classifier. Varying the decision threshold on the classification posterior probability results in the ROC curves shown in Fig. 9 and Fig. 10 for INV terms and OOV terms respectively.

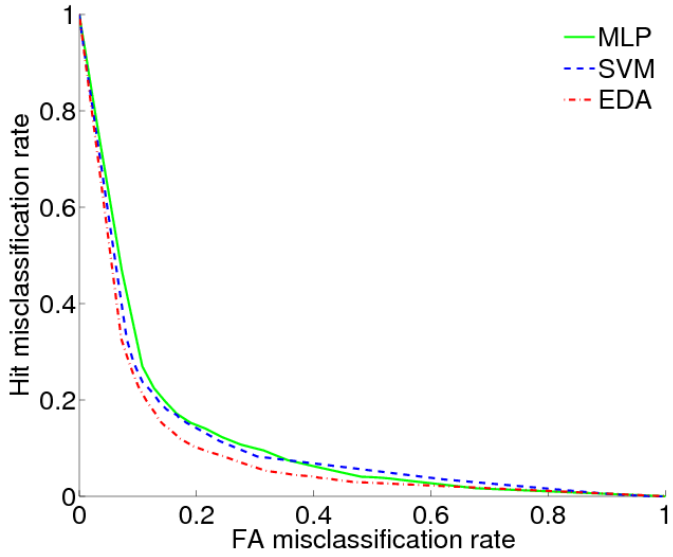

Fig. 9 *ROC curves for INV terms with MLP/SVM/EDA.*

We observe that the EDA model outperforms the MLP and the SVM for both INV terms and OOV terms; especially with OOV terms, EDA substantially outperforms the other two models. This result is fully consistent with

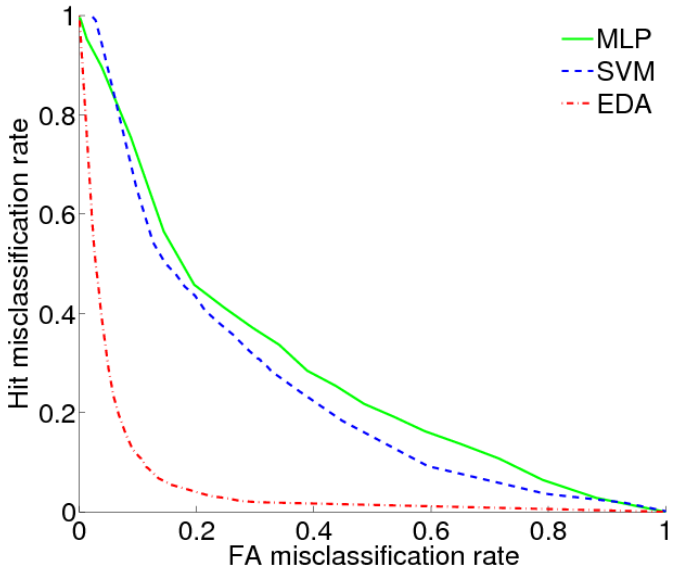

Fig. 10 *ROC curves for OOV terms with MLP/SVM/EDA.*

our analysis in previous sections: on one hand, it confirms our conjecture that OOV terms tend to be more diverse and therefore the decision boundary is more complex in classification; on the other hand, it supports our analysis that EDA leads to more flexible models than MLPs and SVMs and therefore tends to exhibit more advantage on classification tasks with complex decision boundaries, which is the case when classifying detections of OOV terms.

As presented in Section 2, the confidence provided by the discriminative models can be normalized to make an ATWV-oriented decision. This means that the confidence after normalization is more relevant for STD performance. To elucidate the contribution of confidence normalization, we present the ROC curves for classification performance on INV and OOV terms with normalized confidence (Eq. 7) in Fig. 11 and Fig. 12 respectively. Again, we observe different patterns for INV terms and OOV terms: for INV terms, the confidence normalization does not provide any substantial benefit, however, for OOV terms it leads to a performance improvement with all the three models, and in particular, substantially for the MLPs and SVMs. This is to be expected as the normalization technique is proposed to compensate for the diversity in term occurrences, which is clearly more prominent for OOV terms.

Comparing Fig. 10 and Fig. 12, we observe that the improvement achieved with EDA compared to MLP and SVM without confidence normalization diminishes after applying confidence normalization. This suggests that EDA plays the role of confidence normalization to some extent. Specifically, the EDA model, due to its ability to deal with complex decision boundaries, may well take into account the contribution of the term occurrences in modeling

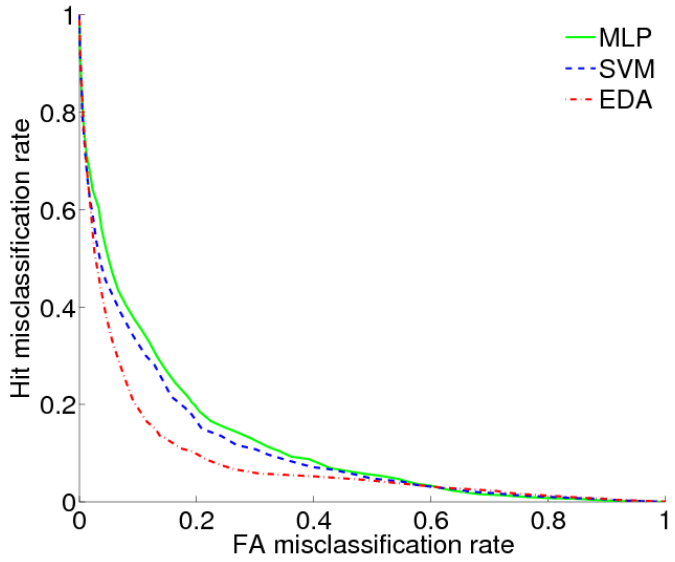

Fig. 11 *ROC curves for INV terms with MLP/SVM/EDA after confidence normalization.*

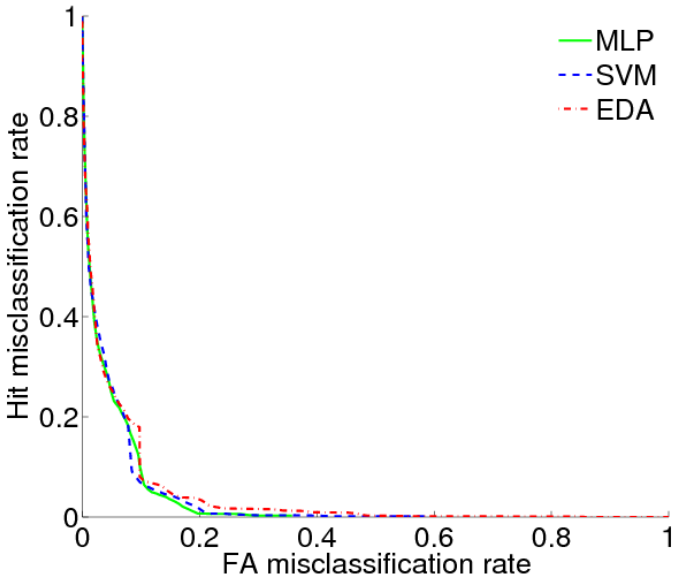

Fig. 12 *ROC curves for OOV terms with MLP/SVM/EDA after confidence normalization.*

and compensate for it, which is precisely the role played by confidence normalization. MLPs and SVMs, although fed with the term occurrence features,

are less capable of modeling them. With confidence normalization applied, the term occurrences are explicitly compensated for all the three models, leading to a less significant EDA advantage. Apart from the implicit normalization that EDA makes, we should note that EDA is far superior for occurrence compensation – it is capable of taking any possible features and assigning them appropriate roles in decision making, by its evolutionary mechanism and the evaluation metric-oriented optimization.

Table 1 presents the equal error rate (EER) with the three discriminative models, with and without confidence normalization. Although the patterns can be observed from the ROC curves, this table provides a direct comparison for different models on different categories of terms with and without normalization. Note that the EER is not proportional to ATWV in STD that we will report in the next section. Firstly, EER is occurrence-averaged while ATWV is term-averaged; secondly, EER considers the detections that have been hypothesized by STD, whereas ATWV considers even the occurrences that are missed by STD; and finally, EER and ATWV reflect system behavior with different hit/FA ratios. Therefore, the smaller EER on OOV terms in Table 1 does not indicate better performance on STD with OOV terms – it just results from the fact that more OOV occurrences tend to be missed by STD and the OOV detections usually involve more false alarms that are easy to classify. Nevertheless, the results we obtained so far, clearly demonstrate that the EDA is more effective than the other two models in classification, and this advantage is more prominent when classifying detections of OOV terms. It is reasonable to expect that this advantage leads to an improvement in the STD performance.

	EER	
Confidence estimator	INV terms	OOV terms
MLP	0.17	0.34
+ conf. norm.	0.20	0.10
SVM	0.17	0.31
+ conf. norm.	0.19	0.09
EDA	0.15	0.11
+ conf. norm.	**0.14**	**0.09**

Table 1 *EER with MLP/SVM/EDA, with and without confidence normalization. "conf. norm." denotes confidence normalization.*

5.3 Spoken term detection

After normalization, the discriminative confidence can be used for STD to make the hit/FA decision according to Eq. 7. The results in terms of ATWV on both INV and OOV terms are shown in Table 2. We observe that the confidence based on EDA outperforms those based on the MLP and SVM for both INV and OOV terms. Paired t-tests show that this improvement is

statistically significant ($p < 0.001$) for OOV terms compared with the SVM and weakly significant ($p < 0.09$) compared with the MLP. For INV terms, the improvement achieved by EDA is insignificant compared with the MLP ($p \approx 0.4$) and hardly significant compared with the SVM ($p \approx 0.1$).

	ATWV	
Confidence estimator	INV terms	OOV terms
MLP	0.5466	0.2952
SVM	0.5434	0.2920
EDA	**0.5500**	**0.2994**

Table 2 *STD performance based on discriminative confidence estimated by the MLP, SVM and EDA with the best result in bold. Results are reported in terms of ATWV, and for both INV and OOV terms. Confidence normalization is applied.*

Applying various decision thresholds leads to the DET curves shown in Fig. 13 and Fig. 14 for INV and OOV terms respectively. We can see that EDA outperforms both the MLP and SVM considerably for OOV terms especially when the FA is low. We suppose that this is because EDA-based confidence is derived from distance to classification boundaries, which is more *normalized* than those derived from intermediate objective functions used by MLPs and SVMs especially in tasks involving complex decision boundaries. Therefore, the confidence obtained from our EDA approach tends to be more robust against threshold variation. For INV terms, EDA does not show an obvious advantage over the other two models. This is consistent with the results on the classification task as well as with the results in terms of ATWV, and strongly supports our hypothesis that EDA is an advanced tool for complex decision tasks and can be employed to boost STD on OOV terms.

5.4 Discussion

We have demonstrated that the EDA approach is more effective than MLPs and SVMs, especially when the decision boundary is complex in classification, and this advantage can be carried to related tasks such as OOV STD. We have attributed this EDA advantage to its capability of compensating for the high diversity among OOV terms. We also demonstrated that the term occurrence rate is among those properties that result in the high diversity. An interesting question that arises in this context is: are there other properties that lead to the OOV diversity? As an extended investigation, we examine the relationship between the term occurrence rate and the relative advantage of EDA.

In Fig. 15 and Fig. 16, the X-axis represents term occurrences and the Y-axis represents the number of terms on which each model achieves the best performance in terms of ATWV. Fig. 15 presents the results on OOV terms and Fig. 16 presents the results on INV terms; confidence normalization has been applied in both cases. We can see that, for OOV terms, the EDA approach clearly outperforms both MLPs and SVMs for a wide range of occurrence

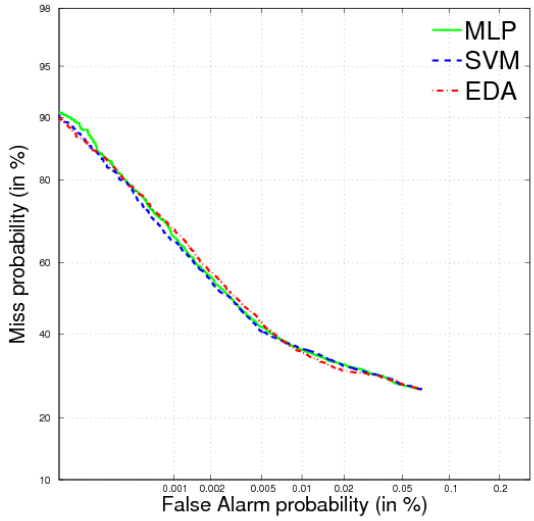

Fig. 13 *DET curves of the STD system with discriminative confidence based on MLP, SVM and EDA. Results are reported for INV terms. Confidence normalization is applied.*

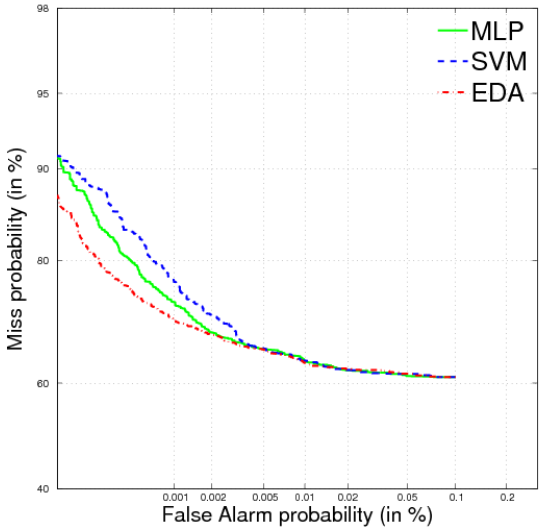

Fig. 14 *DET curves of the STD system with discriminative confidence based on MLP, SVM and EDA. Results are reported for OOV terms. Confidence normalization is applied.*

rates, particularly when the term occurrence rate is low. This on one hand suggests that the EDA-based confidence is more robust to variations in term

occurrences, and is more effective in compensating for low occurrences than MLPs and SVMs, and on the other hand, as EDA wins in almost all term occurrence rates (terms with 9 occurrences is the only exception), suggests that there must be some other properties besides term occurrences (e.g., the diversity in phonetic regulation) that are diverse among OOV terms but are effectively normalized by the EDA approach. For INV terms, we find that different models win at different occurrence rates, supporting our conjecture that INV terms, due to their similar patterns, benefit less from the robustness of the EDA-based confidence measure.

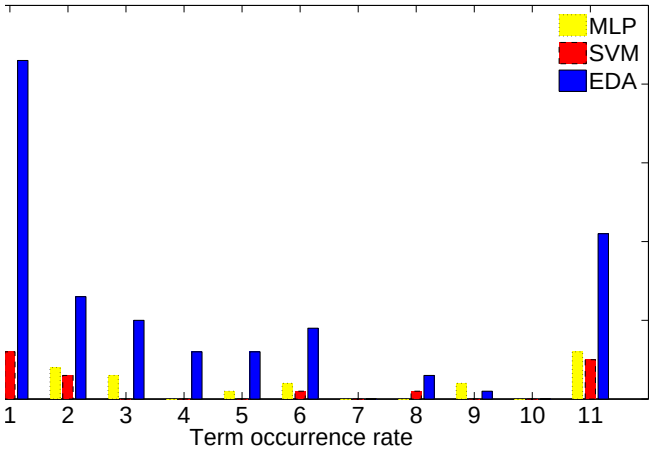

Fig. 15 *Histogram of OOV terms that achieve the best STD performance with each discriminative model. The X-axis represents the term occurrence rate and the Y-axis represents the number of terms that have a particular term occurrence rate and achieve the best STD performance with each model (MLP, SVM or EDA). The X-axis corresponding to* 11 *refers to all the terms which have more than* 10 *occurrences.*

Generally speaking, the power of EDA is largely relevant to the flexible architecture of combining heterogeneous projection functions and decision strategies. Any continuous or discrete projection function and classification strategy can be integrated together, lending EDA the capability to cope with highly complex classification tasks, e.g., those with a piece-wise decision boundary. The cost of the high flexibility is two-fold: first it may place high demand on computation, second it tends to cause serious over-fitting problems. In our experiments, training an EDA model with a fixed MLP structure takes around 160 minutes (only a few minutes are required for the MLP or SVM training). This is still affordable for a task of the same scale as ours. The EERs on the training and evaluation data with EDA are 0.13 and 0.15 respectively (on INV terms). Considering the superior results over the other two models, we can assume no obvious over-fitting involved in the EDA training in our experiments.

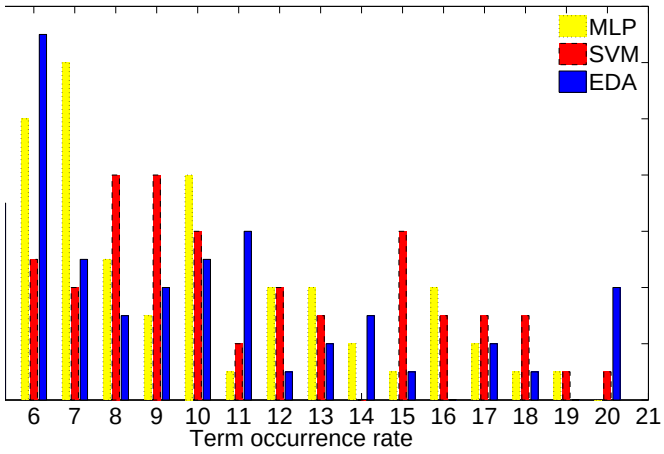

Fig. 16 *Histogram of INV terms that achieve the best STD performance with each discriminative model. The X-axis represents the term occurrence rate and the Y-axis represents the number of terms that have a particular term occurrence rate and achieve the best STD performance with each model (MLP, SVM or EDA).*

When applied to other tasks, however, selection of the EDA structure and its configuration might be non trivial.

Finally, we note that the EDA-based approach is general and can be employed for confidence estimation in a broad range of applications such as word-based STD or ASR output scoring. However, there is some trade-off between the task scale and the improvement that EDA may achieve, since EDA training itself is computationally demanding. Readers can find some results on word-based STD systems in [53], which shows that word-based systems typically generate low false alarm rates, and therefore the performance obtained with the discriminative confidence estimation is less prominent compared to phone-based systems.

6 Conclusions

This paper proposed a new confidence estimation approach based on EDA for spoken term detection. Both the theoretical analysis and the experimental results demonstrate that EDA is more effective than MLPs and SVMs on classification tasks especially those with complex decision boundaries. This advantage can be exploited to improve the hit/FA decision making in STD for OOV terms which are difficult to classify due to their highly diverse properties (ASR error patterns, occurrence rates and confidence distributions) and more risk of converging to local minima in model training. Our experiments confirm that a significant performance improvement can be obtained for OOV terms with EDA-based confidence than with confidence based on SVMs and MLPs,

whilst the improvement for INV terms is relatively marginal. The EDA approach significantly outperforms both MLPs and SVMs in terms of ATWV by a relative improvement of 1.4% and 2.5% respectively. These results validate our analysis that the EDA approach, which is based on a flexible combination of heterogeneous projection and classification functions and using the measurement metric as its objective, is able to deal with complex tasks and ameliorates the risk of local minima, suggesting that the EDA is a better model than the MLP and SVM in dealing with OOV terms in STD. The main drawback of EDA as compared to MLPs and SVMs is the high computational cost in model training, which in our case is about 2.5 hours, much higher than that required to train MLPs and SVMs (just a few minutes). Nevertheless, this is still acceptable for tasks that are not over complex, at least in STD confidence estimation.

Future work will investigate other projection functions and other classifiers to enhance the EDA model. In addition, new fitness functions will be investigated, particularly one that optimizes ATWV directly.

Acknowledgements This work was partially supported by the French Ministry of Industry (Innovative Web call) under contract 09.2.93.0966, 'Collaborative Annotation for Video Accessibility' (ACAV) and by 'The Adaptable Ambient Living Assistant' (ALIAS) project funded through the joint national Ambient Assisted Living (AAL) programme.

References

1. Akbacak, M., Vergyri, D., Stolcke, A.: Open-vocabulary spoken term detection using graphone-based hybrid recognition systems. In: Proc. ICASSP'08, pp. 5240–5243. Las Vegas, USA (2008)
2. Alander, J.T.: Indexed bibliography of genetic algorithms in physical sciences. Report 94-1-PHYS, University of Vaasa, Department of Information Technology and Production Economics (1995)
3. Beyer, H.G., Schwefel, H.P.: Evolution strategies - a comprehensive introduction. Journal of Natural Computing **1**(1), 3–52 (2002)
4. Bisani, M., Ney, H.: Joint-sequence models for grapheme-to-phoneme conversion. Speech Communication **50**(5), 434–451 (2008)
5. Bishop, C.M.: Neural Networks for Pattern Recognition. Oxford University Press (1995)
6. Black, A.W., Lenzo, K., Pagel, V.: Issues in building general letter to sound rules. In: Proc. 3rd ESCA Workshop on Speech Synthesis, pp. 77–80. Jenolan Caves, Australia (1998)
7. Burger, S., MacLaren, V., Yu, H.: The ISL meeting corpus: the impact of meeting type on speech style. In: Proc. ICSLP'02, pp. 301–304. Denver, USA (2002)
8. Can, D., Cooper, E., Sethy, A., White, C., Ramabhadran, B., Saraclar, M.: Effect of pronunciations on OOV queries in spoken term detection. In: Proc. ICASSP'09, pp. 3957–3960. Taipei, Taiwan (2009)
9. Chan, C.A., Lee, L.S.: Unsupervised spoken-term detection with spoken queries using segment-based dynamic time warping. In: Proc. Interspeech'10 (2010)
10. Chang, C.C., Lin, C.J.: LIBSVM: A library for support vector machines (2001)
11. Chen, C.P., Lee, H.Y., Yeh, C.F., Lee, L.S.: Improved spoken term detection by feature space pseudo-relevance feedback. In: Proc. Interspeech'10 (2010)
12. Chen, S.H.: Evolutionary Computation in Economics and Finance. Springer-Verlag New York, Inc., Secaucus, NJ, USA (2002)

13. Cordón, O., Herrera-Viedma, E., López-Pujalte, C., Luque, M., Zarco, C.: A review on the application of evolutionary computation to information retrieval. International Journal of Approximate Reasoning **34**(2–3), 241–264 (2003). Soft Computing Applications to Intelligent Information Retrieval on the Internet
14. Daelemans, W., van den Bosch, A., Zavrel, J.: Forgetting exceptions is harmful in language learning. Machine Learning **34**(1-3), 11–41 (1999)
15. Damper, R., Eastmond, J.: Pronunciation by analogy: Impact of implementational choices on performance. Language and Speech **40**(1), 1–23 (1997)
16. Deligne, S., Yvon, F., Bimbot, F.: Variable-length sequence matching for phonetic transcription using joint multigrams. In: Proc. Eurospeech'95, pp. 2243–2246. Madrid, Spain (1995)
17. Eiben, A.E., Smith, J.E.: Introduction to Evolutionary Computing. SpringerVerlag (2003)
18. Filho, E., De Carvalho, A.: Evolutionary design of MLP neural network architectures. In: Neural Networks, 1997. Proceedings., IVth Brazilian Symposium on, pp. 58 –65 (1997)
19. Fiscus, J.G., Ajot, J., Garofolo, J.S., Doddington, G.: Results of the 2006 spoken term detection evaluation. In: Proc. Workshop on Searching Spontaneous Conversational Speech (SIGIR-SSCS'07). Amsterdam (2007)
20. Fisher, R.A.: The use of multiple measurements in taxonomic problems. Annals of Eugenics **7**, 179–188 (1936)
21. Fogel, G.B., Corne, D.W.: Evolutionary Computation in Bioinformatics. Elsevier (2002)
22. Hain, T., Burget, L., Dines, J., Garau, G., Karafiat, M., Lincoln, M., Vepa, J., Wan, V.: The AMI meeting transcription system: Progress and performance. In: Machine Learning for Multimodal Interaction, vol. 4299/2006, pp. 419–431. Springer Berlin/Heidelberg (2006)
23. Janin, A., Baron, D., Edwards, J., Ellis, D., Gelbart, D., Morgan, N., Peskin, B., Pfau, T., Shriberg, E., Stolcke, A., Wooters, C.: The ICSI meeting corpus. In: Proc. ICASSP'03, pp. 364–367. Hong Kong (2003)
24. Jansen, A., Church, K., Hermansky, H.: Towards spoken term discovery at scale with zero resources. In: Proc. Interspeech'10 (2010)
25. Logan, B., Moreno, P., Thong, J.M.V., Whittaker, E.: An experimental study of an audio indexing system for the web. In: Proc. ICSLP'00, vol. 2, pp. 676–679. Beijing, China (2000)
26. Mamou, J., Ramabhadran, B.: Phonetic query expansion for spoken document retrieval. In: Proc. Interspeech'08, pp. 2106–2109. Brisbane, Australia (2008)
27. Mamou, J., Ramabhadran, B., Siohan, O.: Vocabulary independent spoken term detection. In: Proc. ACM-SIGIR'07, pp. 615–622 (2007)
28. Mantere, T., Alander, J.T.: Evolutionary software engineering, a review. Appl. Soft Comput. **5**, 315–331 (2005)
29. Martin, A., Doddington, G., Kamm, T., Ordowski, M., Przybocki, M.: The DET curve in assessment of detection task performance. In: Proc. Eurospeech'97, vol. 4, pp. 1895–1898,. Rhodes, Greece (1997)
30. Meng, S., Yu, P., Liu, J., Seide, F.: Fusing multiple systems into a compact lattice index for Chinese spoken term detection. In: Proc. ICASSP'08, pp. 4345–4348. Las Vegas, USA (2008)
31. Michalewicz, Z.: Evolutionary Algorithms in Engineering Applications, 1st edn. Springer-Verlag New York, Inc., Secaucus, NJ, USA (1997)
32. Miller, D.R.H., Kleber, M., Kao, C.L., Kimball, O., Colthurst, T., Lowe, S.A., Schwartz, R.M., Gish, H.: Rapid and accurate spoken term detection. In: Proc. Interspeech'07, pp. 314–317. Antwerp, Belgium (2007)
33. Mitchell, M., Taylor, C.E.: Evolutionary computation: An overview. Annual Review of Ecology and Systematics **30**(1), 593–616 (1999)
34. Motlicek, P., Valente, F., Garner, P.: English spoken term detection in multilingual recordings. In: Proc. Interspeech'10 (2010)
35. NIST: The spoken term detection (STD) 2006 evaluation plan. National Institute of Standards and Technology (NIST), Gaithersburg, MD, USA, 10 edn. (2006). URL http://www.nist.gov/speech/tests/std

36. Parada, C., Sethy, A., Dredze, M., Jelinek, F.: A spoken term detection framework for recovering out-of-vocabulary words using the web. In: Proc. Interspeech'10 (2010)
37. Parada, C., Sethy, A., Ramabhadran, B.: Balancing false alarms and hits in spoken term detection. In: Proc. ICASSP'10 (2010)
38. Rocha, M., Cortez, P., Neves, J.: Evolutionary design of neural networks for classification and regression. In: B. Ribeiro, R. Albrecht, A. Dobnikar, D. Pearson, N. Steele (eds.) ADAPTIVE AND NATURAL COMPUTING ALGORITHMS, pp. 304–307. Springer-Verlag Wien (2005)
39. Rocha, M., Cortez, P., Neves, J.: Evolution of neural networks for classification and regression. Neurocomput. **70**, 2809–2816 (2007)
40. Schneider, D., Mertens, T., Larson, M., Kohler, J.: Contextual verification for open vocabulary spoken term detection. In: Proc. Interspeech'10 (2010)
41. Sierra, A., Echeverría, A.: Neural networks trained by distance to means. WSEAS Transactions on Information Science and Applications **2**(9), 1446–1453 (2005)
42. Sierra, A., Echeverría, A.: Evolutionary discriminant analysis. IEEE Transactions on Evolutionary Computation **10**(1), 81–92 (2006)
43. Smith, S.L., Cagnoni, S.: Genetic and Evolutionary Computation: Medical applications. Wiley (2010)
44. Szöke, I., Burget, L., Černocký, J., Fapšo, M.: Sub-word modeling of out of vocabulary words in spoken term detection. In: Proc. IEEE Workshop on Spoken Language Technology (SLT'08), pp. 273–276. Goa, India (2008)
45. Szöke, I., Fapšo, M., Burget, L., Černocký, J.: Hybrid word-subword decoding for spoken term detection. In: Proc. Speech search workshop at SIGIR (SSCS'08). Association for Computing Machinery, Singapore (2008)
46. Szöke, I., Fapšo, M., Karafiát, M., Burget, L., Grézl, F., Schwarz, P., Glembek, O., Matějka, P., Kontár, S., Černocký, J.: BUT system for NIST STD 2006 - English. In: Proc. NIST Spoken Term Detection Evaluation workshop (STD'06). National Institute of Standards and Technology, Maryland, USA (2006)
47. Szöke, I., Fapšo, M., Karafiát, M., Burget, L., Grézl, F., Schwarz, P., Glembek, O., Matějka, P., Kopecký, J., Černocký, J.: Spoken term detection system based on combination of LVCSR and phonetic search. In: Machine Learning for Multimodal Interaction, *Lecture Notes in Computer Science*, vol. 4892/2008, pp. 237–247. Springer Berlin / Heidelberg (2008)
48. Taylor, P.: Hidden Markov models for grapheme to phoneme conversion. In: Proc. Interspeech'05, pp. 1973–1976. Lisbon, Portugal (2005)
49. Thambiratmann, K., Sridharan, S.: Rapid yet accurate speech indexing using dynamic match lattice spotting. IEEE Transactions on Audio, Speech, and Language Processing **15**(1), 346–357 (2007)
50. Vergyri, D., Shafran, I., Stolcke, A., Gadde, R.R., Akbacak, M., Roark, B., Wang, W.: The SRI/OGI 2006 spoken term detection system. In: Proc. Interspeech'07, pp. 2393–2396. Antwerp, Belgium (2007)
51. Vergyri, D., Stolcke, A., Gadde, R.R., Wang, W.: The SRI 2006 spoken term detection system. In: Proc. NIST spoken term detection workshop (STD 2006). Gaithersburg, Maryland, USA (2006)
52. Wallace, R., Vogt, R., Baker, B., Sridharan, S.: Optimising Figure of Merit for phonetic spoken term detection. In: Proc. ICASSP'10 (2010)
53. Wang, D.: Out-of-vocabulary spoken term detection. Ph.D. thesis, The Center for Speech Technology Research, Edinburgh University (2009)
54. Wang, D., King, S., Frankel, J.: Stochastic pronunciation modelling for spoken term detection. In: Proc. Interspeech'09, pp. 2135–2138. Brighton, UK (2009)
55. Wang, D., King, S., Frankel, J.: Stochastic pronunciation modeling for out-of-vocabulary spoken term detection. IEEE Transactions on Audio, Speech, and Language Processing **19**(4), 688–698 (2010)
56. Wang, D., King, S., Frankel, J., Bell, P.: Term-dependent confidence for out-of-vocabulary term detection. In: Proc. Interspeech'09, pp. 2139–2142. Brighton, UK (2009)
57. Wang, D., King, S., Frankel, J., Vipperla, R., Evans, N., Troncy, R.: Direct posterior confidence estimation for out-of-vocabulary spoken term detection. Submitted to ACM Transactions on Information Systems

58. Watson, D.: Death Sentence, The Decay of Public Language. Knopf, Sydney (2003)
59. Wessel, F., Macherey, K., Schlüter, R.: Using word probabilities as confidence measures. In: Proc. ICASSP'98, vol. 1, pp. 225–228. Seattle, Washington, USA (1998)