Neural Computing and Applications (2021) 33:9733–9750
https://doi.org/10.1007/s00521-021-05741-0

ORIGINAL ARTICLE

Insight into an unsupervised two-step sparse transfer learning algorithm for speech diagnosis of Parkinson's disease

Yongming Li[1] · Xinyue Zhang[1] · Pin Wang[1] · Xiaoheng Zhang[1,2] · Yuchuan Liu[1]

Received: 1 August 2020 / Accepted: 16 January 2021 / Published online: 9 February 2021

Abstract

Speech diagnosis of Parkinson's disease (PD) as a non-invasive and simple diagnosis method is particularly worth exploring. However, the number of samples of speech-based PD is relatively small, and there exist discrepancies in the distribution between subjects. In order to solve the two problems, a novel unsupervised two-step sparse transfer learning is proposed in this paper to tackle with PD speech diagnosis. In the first step, convolution sparse coding with the coordinate selection of samples and features is designed to learn speech structure from the source domain to replenish sample information of the target domain. In the second step, joint local structure distribution alignment is designed to maintain the neighbor relationship between the respective samples of the training set and test set, and reduce the distribution difference between the two domains at the same time. Two representative public PD speech datasets and one real-world PD speech dataset were exploited to verify the proposed method on PD speech diagnosis. Experimental results demonstrate that each step of the proposed method has a positive effect on the PD speech classification results, and it also delivers superior performance over the existing relative methods.

Keywords Speech diagnosis · Parkinson's disease · Two-step sparse transfer learning · Convolution sparse coding · Domain adaptation

1 Introduction

Parkinson's disease (PD) is the second most common degenerative disorder of the nervous system, occurs mostly in the elderly population, and generally deteriorates over time [1]. According to the research, more than 5% of PD is hereditary [2]. With the aging population trend, the number of cases increased year by year [3]. So far, there is no way to cure or prevent PD, but this disease can be controlled through early diagnosis and treatment [4, 5]. Thus, early diagnosis is critical to improve the patient's quality of life and prolong their lives [6].

Speech disorder is one of the typical symptoms of PD which is commonly called Parkinson's dysarthria [7–9]. Several studies in the literature have described the speech impairments of PD patients in terms of phonation, articulation, and prosody [10–12]. Along with these three aspects of speech, intelligibility is also deteriorated in PD patients causing loss of communication abilities and social isolation, especially at advanced stages of the disease [13]. Therefore, it is of great scientific value and practical significance to further study PD diagnostic ability based on speech datasets, since utilizing speech data can help develop a simple, fast, and non-invasive early PD diagnostic method. The literature shows that a sizable number of researchers have made many attempts to classify people correctly as either PD patients or healthy people based on

✉ Yongming Li
yongmingli@cqu.edu.cn

✉ Xiaoheng Zhang
zhangxh@cqu.edu.cn; 00816@cqdd.cq.cn

Xinyue Zhang
20181202035t@cqu.edu.cn

Pin Wang
wangpin@cqu.edu.cn

Yuchuan Liu
20191201413@cqu.edu.cn

[1] School of Microelectronics and Communication Engineering, Chongqing University, Chongqing 400030, China

[2] Chongqing Radio and TV University, Chongqing 400052, China

speech data. They are mostly based on different extracted features, feature selection/transformation methods [8, 14–29], or classifiers [14–16, 20–22, 27, 30–35] to maximize the accuracy of classification of Parkinson's disease. As for feature extraction, the PD speech feature data primarily include pitch type, energy type, speed type, and content type [8, 14, 15]. About feature selection/transformation, the frequently used algorithms are neural network (NN) based [16–18], principal component analysis (PCA) [19–21], serial search based [14, 19, 21], evolutionary based [18, 22, 23], p value based [15, 24], relevance based [25–27], entropy based [28], and LDA based [29] methods. As for classifier design, support vector machine (SVM) [14–16] and k-nearest neighbor (KNN) [14, 15, 20] are two most commonly used classifiers. Others are random forest (RF) [14, 27], Bayesian network [30, 31], discrimination algorithm (DA) [21, 32], probabilistic neural network (PNN) [33–35], decision tree [21, 34], non-nested generalized exemplars (NNge) [35], and so on. Although there are many classification algorithms for PD diagnosis, the predicted results still leave much room to improve.

It is worth noting that all the methods above are based on the classification of the original speech dataset and do not take the small sample problem of the dataset into consideration. Transfer learning has the potential to address these problems [36], and the studies [14, 16, 37] confirm the effectiveness of transfer learning in the diagnosis of PD. However, these transfer learning methods only pay attention to the distribution difference between the source domain and the target domain and ignore the difference between the data diversity of the target domain. Some researchers [38–40] have shown that the training set and test set can be regarded as different domains to reduce data distribution difference, but there are few studies in the field of PD speech diagnosis. All the transfer learning methods mentioned above belong to one-step transfer learning and do the transfer from different datasets. The two-step transfer learning methods have achieved significant results in some areas recently [41–43]. For instance, Sakurai et al. [41] achieved semantic plant segmentation by two-step domain adaptation: firstly, adaptation is from a large amount of labeled data to a major category and then adapted category adaptation from the major category to a minor category; An et al. [42] realized age-related macular degeneration diagnosis based on twice transfer of models: firstly, a pre-trained VGG16 model was used, and then, the fine-tuned model in the first step was to transfer learned again to distinguish the images; Similar to G, Zhang et al. [43] utilized two-step transfer learning to detect COVID-19 based on model-transfer, but in different models. Specifically, all the above two-step transfer approaches are based on images and designed differently for different data characteristics and tasks, but are not considered in PD speech recognition. Moreover, there exist discrepancies in the distribution between subjects of PD within single dataset, but the existing methods did not considered this point.

In order to solve the problems above, the unsupervised two-step sparse transfer learning (TSTL) is proposed in this paper in PD's speech diagnosis. In the first step, convolution sparse coding learning with the coordinate selection of samples and features (CSC&SF) is proposed to supplement the structure information of PD speech, as for the small samples. And in the second step, due to the discrepancies of subjects, joint local structure distribution alignment (JLSDA) is designed to realize distribution alignment of the training set and test set and retain its original structure.

To sum up, the contributions and innovations of this paper are mainly described as follows:

(1) A novel two-step transfer learning algorithm, called TSTL is proposed for the classification of PD speech data. The method can help learn useful information from large unlabeled speech data, align the distribution of training set and test set, and retain the original structure between samples at the same time.
(2) Transfer learning between different datasets and transfer learning between the training set and test set are combined to construct an unsupervised two-step sparse transfer learning algorithm for the first time.
(3) For the first time in the same speech PD dataset, the problem of individual differences among samples is considered as the problem of distribution differences between the training set and test set of PD speech data.

The rest of this paper is organized as follows. Section 2 reviews prior works that are related to proposed method. Section 3 introduces the theoretical part of the proposed algorithm. Section 4 describes the experiments to verify the effectiveness of TSTL and each step of it. Section 5 is the discussions and conclusions about this proposed method and future work.

2 Related works

The proposed method TSTL is a two-step transfer learning method applied in PD speech diagnosis. Thus, this section presents the detail of the prior works on two parts of TLSLT.

The first step transfer learning is related to convolution sparse coding (CSC) [44–46], which has great unsupervised sparse learning ability and can find out the implicit structures and patterns in the input data effectively. And CSC can extract the features reflecting the structures and

relationship between features and samples, while controlling the number of the features. The transfer learning can be combined with sparse coding [47] to extract more valued information from the public speech datasets, thereby solving the small sample problem and finding out the structures and patterns implicit in the input data at the same time. Due to the small size of PD speech datasets, it is difficult to expand the amount of data. Then, enrich speech structure information by CSC become a valid and feasible way.

The second step transfer learning is concerned with domain adaptation (DA) which aims at transferring shared knowledge across different, but related tasks or domains [48]. The common practice for unsupervised domain adaptation (UDA) is to minimize the discrepancy between domains to obtain domain-invariant features [49–53] or learn more discriminative features, while performing domain alignment [54–58]. And there are no labeled instances in the target domain. According to whether the feature space of the source domain and the target domain are similar and have the same-dimensionality, UDA can be divided into homogeneous unsupervised domain adaptation (HoUDA) and heterogeneous unsupervised domain adaptation (HeUDA) [59]. Due to the small size of PD speech datasets, like most domain adaptation models, TSTL is focused on HoUDA.

And according to whether generalize deep convolutional neural network to the domain adaptation scenario, UDA can be divided into traditional UDA and deep UDA methods. As for traditional UDA methods, transfer component analysis (TCA) [48] and joint distribution adaptation (JDA) [60] are based on maximum mean discrepancy (MMD), geodesic flow kernel (GFK) [61] proposes to learn the geodesic flow kernel between domains in manifold space, manifold embedded distribution alignment (MEDA) [62] learns a domain-invariant classifier, correlation alignment (CORAL) [39] adjusts the covariance of different domains. As for deep UDA methods, deep adaptation networks (DAN) [63] applies MK-MMD to adapt multilayer feature, joint adaptation network (JAN) [64] adds joint distribution on the basis of DAN, and inspired by GANs, the single-adversarial model domain-adversarial neural network (DANN) [65] and multi-adversarial model conditional domain adversarial networks (CDANs) [66] are proposed. Although those UDA show strong robustness and generalization among datasets in various fields, the most are applied in image classification and not match the data characteristics of PD speech datasets. The typical UDA methods are compared with proposed approach in Sect. 4.

3 The proposed method

3.1 Problem formulation

The PD speech datasets have the typical characteristics of small samples, which make the training sample insufficient, easily lead to overfitting, and worsens the generalization ability of the classification model. However, there are few relevant methods for diagnosis of PD to deal with the problems above, especially in the field of speech diagnosis of PD. Besides, most algorithms do not consider the effect of differences between Parkinson's subjects. To solve these problems, a two-step sparse transfer learning idea is proposed here. In the first transfer step, the goal is to learn useful information from the public speech data (source domain) and transfer it to the PD speech dataset (target domain) to increase the generalization ability of the PD classification model. The purpose of the second transfer step is to reduce discrepancies by aligning the Parkinson data distribution from training subjects and test subjects. So training subjects are regarded as the source domain, and test subjects are the target domain in the second transfer step. Besides, the original structure between samples is also retained in this transfer step.

In the first transfer step, PD speech dataset is target domain dataset $\boldsymbol{F} = \left[\vec{\boldsymbol{F}}_1, \vec{\boldsymbol{F}}_2, \ldots, \vec{\boldsymbol{F}}_G\right]^{\mathrm{T}}$, where $\vec{\boldsymbol{F}}_i = [f_{i1}, f_{i2}, \ldots, f_{iN}]$, $1 \le i \le G$, partitioned matrix on subjects $\boldsymbol{F} = \left[\tilde{\boldsymbol{F}}_1, \tilde{\boldsymbol{F}}_2, \ldots, \tilde{\boldsymbol{F}}_M\right]^{\mathrm{T}}$, $\tilde{\boldsymbol{F}}_i = \begin{bmatrix} f_{11} & f_{12} & \cdots & f_{1N} \\ f_{21} & f_{22} & \cdots & f_{2N} \\ \cdots & \cdots & \cdots & \cdots \\ f_{G_0 1} & f_{G_0 2} & \cdots & f_{G_0 N} \end{bmatrix}$, $1 \le i \le M$. The total number of samples is G, number of features per sample is N, all samples belong to M subject, that is, the number of samples included in each subject is: $G_0 = G/M$. Before being used as the source dataset in the first transfer step, the public speech dataset is extended to a larger scale by injecting different SNR and different types of noise. Extended dataset is $\boldsymbol{S}' = \left[\vec{\boldsymbol{S}}'_1, \vec{\boldsymbol{S}}'_2, \ldots, \vec{\boldsymbol{S}}'_J\right]^{\mathrm{T}}$, and $\vec{\boldsymbol{S}}'_j = \varphi\left(\vec{\boldsymbol{S}}_j, \vec{\boldsymbol{N}}_j, SNR_j\right)$, where $\vec{\boldsymbol{S}}_j$ is the original speech signal from the public data set, $\vec{\boldsymbol{N}}_j$ are different types of noise signals, $\varphi(\cdot)$ is a function of that adjustment of the types of noise and signal-to-noise ratio (SNR). Features are extracted from the extended data sets and form new feature dataset $\boldsymbol{Y} = \left[\vec{\boldsymbol{Y}}_1, \vec{\boldsymbol{Y}}_2, \ldots, \vec{\boldsymbol{Y}}_L\right]^{\mathrm{T}}$, as the source domain dataset, where $\vec{\boldsymbol{Y}}_i = \left[\xi_1\left(\vec{\boldsymbol{S}}'_i\right), \xi_2\left(\vec{\boldsymbol{S}}'_i\right), \ldots, \xi_N\left(\vec{\boldsymbol{S}}'_i\right)\right], 1 \le i \le L$, the feature extraction method in [15] was adopted to extract N different features of the signal. Then, $\boldsymbol{Y}$ can be expressed as $\boldsymbol{Y} = \left[\tilde{\boldsymbol{Y}}_1, \tilde{\boldsymbol{Y}}_2, \ldots, \tilde{\boldsymbol{Y}}_L\right]^{\mathrm{T}}$.The total

number of feature samples is L, $\tilde{\boldsymbol{Y}}_i$ is a two-dimensional $G_0 \times N$ block matrix, $\vec{\boldsymbol{Y}}_i$ is a sparse dictionary learning training sample, and $\tilde{\boldsymbol{Y}}_i$ is convolution kernels sparse learning training samples.

In the second transfer step, a domain $\boldsymbol{Q}$ is composed of a d-dimensional feature space $\boldsymbol{X}'$ and a marginal probability distribution $P(x')$, the source data is denoted as $\boldsymbol{X}'_S = [x'_{S_1}, x'_{S_2}, \ldots, x'_{S_{N_S}}]^{\mathrm{T}} \in \Re^{N_S \times L}$, the target data is denoted as $X'_T = [x'_{T_1}, x'_{T_2}, \ldots, x'_{T_{N_T}}]^{\mathrm{T}} \in \Re^{N_T \times L}$, N_S, N_T are the number of samples in the source and target domains, respectively. All subscripts S represent samples from the source domain or transformed data from the source domain, same for T. The label vector of data is denoted $Y'_S = [y'_{S_1}, y'_{S_2}, \ldots, y'_{S_{N_S}}]^{\mathrm{T}} \in \Re^{N_S}$. C is the number of classes. The symbol $\|\cdot\|_{\mathrm{H}}$ is the reproducing kernel Hilbert space (RKHS) norm. $\mathrm{tr}(\cdot)$ denotes the trace operator and $N_{K(\cdot)}$ denotes the k nearest neighbors operator.

3.2 Brief description of proposed algorithm

The proposed TSTL based on PD speech data consists of two major steps: CSC&SF and JLSDA. In the first step (CSC&SF), its purpose is to learn useful information from public speech data (source domain) and transfer it to the target domain. First, the public speech dataset is expanded with noise injection into a larger one. Second, the features are extracted from the data, thereby constructing a speech feature dataset as the source domain. Then, the CSC learning method is carried out on the source domain datasets, and the kernel matrix is obtained. Based on the kernels, the target domain dataset is encoded to calculate the feature maps, and they are normalized to construct the norm feature map matrix. Row vectors of the same subject are expanded into a one row vector; and based on the Relief algorithm [67], the most effective features can be chosen to reduce the complexity of classification and constitute a new target dataset. In the second step (JLSDA), its purpose is to align the learned Parkinson data distribution and retain its original structure. The training set is looked as the new source domain, and the test set is looked as the new target domain. Both parts are mapped into a public manifold space through the JLSDA method. Finally, the refreshed training set and test set are put into the subsequent classifier for prediction.

3.3 First step transfer (FT)—CSC&SF

In CSC, given G training samples $\{\boldsymbol{x}_g\}_{g=1}^{G}$, the convolution kernel group is learned by minimizing the objective function $\{\boldsymbol{d}_k\}_{i=1}^{K}$ as follows.

$$\arg\min_{\boldsymbol{e},\boldsymbol{d}} \frac{1}{2}\sum_{g=1}^{M}\left\|\boldsymbol{x}_g - \sum_{k=1}^{K}\boldsymbol{d}_k * e_{g,k}\right\|_2^2 + \eta\sum_{g=1}^{G}\sum_{k=1}^{K}\left\|e_{g,k}\right\|_1$$
$$\text{s.t.}\ \ \|\boldsymbol{d}_k\|_2^2 \le 1,\ \ \forall k = \{1,\ldots,K\}$$

where $\boldsymbol{x}_g = \tilde{\boldsymbol{Y}}_g$ is $G_0 \times N$ block matrix, $e_{g,k}$ is $G_0 \times N$ feature map matrix, approximate the $\boldsymbol{x}_g$ by convolving with the corresponding convolution kernel $\boldsymbol{d}_k$, the notation $*$ denotes the two-dimensional convolution, and η is the regularization factor greater than zero, the solution to the above optimization problems are based on the fundamental classical framework alternating direction method of multipliers (ADMM) [68].

The formula (1) may be re-expressed as

$$\arg\min_{\boldsymbol{e}} \frac{1}{2}\|\boldsymbol{D}\boldsymbol{e} - \boldsymbol{x}\|_2^2 + \eta\|\boldsymbol{e}\|_1,\ \ \text{s.t.}\|\boldsymbol{d}_k\|_2^2 \le 1$$

where $\sum_{k=1}^{K}\boldsymbol{d}_k * e_{g,k} = \boldsymbol{D}\boldsymbol{e}$, $\boldsymbol{D} = [D_1\, D_2 \ldots D_K]$ is the corresponding vectorizable convolution operator of $[\boldsymbol{d}_1\, \boldsymbol{d}_2 \ldots \boldsymbol{d}_K]$, $\boldsymbol{e} = \left[e_1^{\mathrm{T}}\ e_2^{\mathrm{T}} \ldots e_K^{\mathrm{T}}\right]^{\mathrm{T}}$ is feature map vector.

The solution can be divided into the following two processes:

Fixing convolution kernel to solve the feature maps, the formula (2) can be expressed as follows.

$$\arg\min_{\boldsymbol{e},\boldsymbol{b}} \frac{1}{2}\|\boldsymbol{D}\boldsymbol{e} - \boldsymbol{x}\|_2^2 + \eta\|\boldsymbol{b}\|_1,\ \ \text{s.t.}\ \ \boldsymbol{e} - \boldsymbol{b} = 0$$
$$\theta_1(\boldsymbol{e}) = \frac{1}{2}\|\boldsymbol{D}\boldsymbol{e} - \boldsymbol{x}\|_2^2$$

which can be solved via ADMM iterations

$$\begin{aligned}
\boldsymbol{e}^{(j+1)} &= \arg\min_{\boldsymbol{e}}\left\{\theta_1(\boldsymbol{e}) + \frac{\rho}{2}\left\|\boldsymbol{e} - \boldsymbol{b}^j + \boldsymbol{u}^j\right\|_2^2\right\} \\
&= \arg\min_{\boldsymbol{e}} \frac{1}{2}\|\boldsymbol{D}\boldsymbol{e} - \boldsymbol{x}\|_2^2 + \frac{\rho}{2}\left\|\boldsymbol{e} - \boldsymbol{b}^j + \boldsymbol{u}^j\right\|_2^2 \\
\boldsymbol{b}^{(j+1)} &= \arg\min_{\boldsymbol{b}}\left\{\theta_2(\boldsymbol{b}) + \frac{\rho}{2}\left\|\boldsymbol{e}^{(j+1)} - \boldsymbol{b} + \boldsymbol{u}^j\right\|_2^2\right\} \\
&= \arg\min_{\boldsymbol{b}} \eta\|\boldsymbol{b}\|_1 + \frac{\rho}{2}\left\|\boldsymbol{e}^{(j+1)} - \boldsymbol{b} + \boldsymbol{u}^j\right\|_2^2 \\
\boldsymbol{u}^{(j+1)} &= \boldsymbol{u}^j + \boldsymbol{e}^{(j+1)} - \boldsymbol{b}^{(j+1)}
\end{aligned}$$

Fixing feature map to solve the convolution kernel, the formula (2) can be expressed as follows.

$$\arg\min_{\boldsymbol{d},\boldsymbol{c}} \frac{1}{2}\|\boldsymbol{E}\boldsymbol{d} - \boldsymbol{x}\|_2^2,\quad \text{s.t.}\ \ \|\boldsymbol{c}_k\|_2^2 \le 1\ \ \text{and}\ \ \boldsymbol{d} - \boldsymbol{c} = 0$$

In (5), $\theta_1(\boldsymbol{d}) = \frac{1}{2}\|\boldsymbol{E}\boldsymbol{d} - \boldsymbol{x}\|_2^2$, $\theta_2(\boldsymbol{c})$ are the indicator function of convex set $\|\boldsymbol{c}_k\|_2^2 \le 1$, in (6), $\mathrm{prox}(\cdot)$ compute proximal operator, which can be solved via ADMM iterations

$$\begin{aligned}\boldsymbol{d}^{(j+1)} &= \arg\min_{\boldsymbol{d}}\left\{\theta_1(\boldsymbol{d}) + \frac{\rho}{2}\left\|\boldsymbol{d} - \boldsymbol{c}^j + \boldsymbol{v}^j\right\|_2^2\right\} \\ &= \arg\min_{\boldsymbol{d}}\frac{1}{2}\|\boldsymbol{E}\boldsymbol{d} - \boldsymbol{x}\|_2^2 + \frac{\rho}{2}\left\|\boldsymbol{d} - \boldsymbol{c}^j + \boldsymbol{v}^j\right\|_2^2 \\ \boldsymbol{c}^{(j+1)} &= \text{prox}_{\theta_2(\boldsymbol{c})}(\boldsymbol{d}^{(j+1)} + \boldsymbol{v}^j) \\ \boldsymbol{v}^{(j+1)} &= \boldsymbol{v}^j + \boldsymbol{d}^{(j+1)} - \boldsymbol{c}^{(j+1)}\end{aligned}$$

Finally, the set of sparse convolution kernel $[\boldsymbol{d}'_1, \boldsymbol{d}'_2, \ldots, \boldsymbol{d}'_k]$ is obtained by alternating iteration. In order to transform the feature matrix $\boldsymbol{E}$ into one row vector, $\boldsymbol{E}$ is extended to $\boldsymbol{B}$ as follows:

$$\boldsymbol{B} = \begin{bmatrix}\vec{\boldsymbol{B}}_1 \\ \vec{\boldsymbol{B}}_2 \\ \vdots \\ \vec{\boldsymbol{B}}_M\end{bmatrix} = \begin{bmatrix}\textbf{RESHAPE}(\boldsymbol{E}_1 \quad H_0 \times N \quad 1 \times N') \\ \textbf{RESHAPE}(\boldsymbol{E}_2 \quad H_0 \times N \quad 1 \times N') \\ \vdots \\ \textbf{RESHAPE}(\boldsymbol{E}_M \quad H_0 \times N \quad 1 \times N')\end{bmatrix} = \begin{bmatrix}\gamma_{11} & \gamma_{12} & \cdots & \gamma_{1N_0} \\ \gamma_{21} & \gamma_{22} & \cdots & \gamma_{2N_0} \\ \cdots & \cdots & \cdots & \cdots \\ \gamma_{M1} & \gamma_{M2} & \cdots & \gamma_{MN_0}\end{bmatrix}$$

where the feature extension of CSC expands H_0 row vectors $\boldsymbol{E}_i$ of the same subject into one row vector; then normalize $\boldsymbol{B}$ to obtain $\boldsymbol{B}'$ and based on the Relief algorithm, the weight $\vec{\boldsymbol{W}} = [w_1\, w_2 \ldots w_{N_0}]$ of every features can be obtained. By setting number R, the most effective features can be chosen to reduce the complexity of classification and constitute a new target dataset $\boldsymbol{X}'$. The pseudo-code description of the CSC&SF algorithm from Public datasets shown as follows.

3.4 Second step transfer (ST)—JLSDA

In this section, we propose to adapt distribution and keep the structure between samples by finding a public manifold space of source domain (training set) and target domain (test set):

$$\min\left[\left\|\frac{1}{n_S}\sum_{i=1}^{n_S}\phi(x_{S_i}) - \frac{1}{n_T}\sum_{j=1}^{n_T}\phi(x_{T_j})\right\|_{\mathrm{H}}^2 + \frac{1}{2}\sum_{m,n}\|\phi(x_{S_m}) - \phi(x_{S_n})\|_{\mathrm{H}}^2 S_{mn} + \frac{1}{2}\sum_{p,q}\left\|\phi(x_{T_p}) - \phi(x_{T_q})\right\|_{\mathrm{H}}^2 S_{pq}\right]$$

According to the key assumption in most unsupervised domain adaptation methods, $P \neq Q$, but $P(Y_S|X_S) = Q(Y_T|X_T)$ [48]. In fact, this refers to minimizing the distance, which is the first part of the formula. The rest parts describe the relationships between samples in source or target domain. Figure 1 shows the main idea of JLSDA method. Different colors denote different domains and different shape denotes different classes. (a) shows the original data distribution of the source domain (training set) and target domain (testing set); (b) presents the relationship between the samples and the domains after the alignment of the source and target domain distributions, but the neighborhood structure relationship between samples is broken by only aligning the distribution, thus affecting the classification effect of classifier; (c) shows the samples' relationship and domain's distribution after JLSDA, all samples still maintain the original neighborhood relationship, while aligning the domain as (a) and this can be better for classification. The proposed method is described later.

Algorithm 1: Convolution Sparse Coding Algorithm with coordinated selection of samples and features (CSC&SF)

Input: Public dataset $\vec{S}$, target domain dataset F, total sample size H, number of features per sample N, number of subjects M

Output: new target domain dataset $\boldsymbol{X}'$

Procedure:

1: Add different types of noise with different signal-to-noise ratios to the dataset and extend it to the set $\boldsymbol{S}$;
2: Extract the features of the speech samples from $\boldsymbol{S}$ and construct the feature dataset, which is the source domain dataset $\mathbf{Y}$;
3: **for** iteration = 1…iter_num **do**
4: **for** iteration1 = 1…iter_num1 **do**
5: According to formula (4), fix convolution kernel to solve the feature maps.
6: **end for**
7: **for** iteration2 = 1…iter_num2 **do**
8: According to formula (6), fix feature map to solve the convolution kernel.
9: **end for**
10:Transform the feature matrix $\boldsymbol{E}$ into $\boldsymbol{B}$,normalize $\boldsymbol{B}$ to get $\boldsymbol{B}'$
11: According to Relief, select R features with the greatest weight to construct the new target set $\boldsymbol{X}'$

3.4.1 Marginal distribution adaptation

Since there are no label in target, using the assumption that $\boldsymbol{Q}^1 \neq \boldsymbol{Q}^2$ domain in this paper, but there exists a transformation ϕ such that $P(\phi(\mathsf{X}_S)) \approx P(\phi(\mathsf{X}_T))$ and $P(\mathsf{Y}_S|\phi(\mathsf{X}_S)) \approx P(\mathsf{Y}_T|\phi(\mathsf{X}_T))$. A major computational issue is to reduce the distribution difference by explicitly minimizing proper distance measure. The distance between two distributions $\boldsymbol{Q}^1$ and $\boldsymbol{Q}^2$ can be empirically measured by MMD (Maximum Mean Discrepancy) [48, 60, 69], being written as:

$$\mathrm{Dist}\left(\boldsymbol{X}'_S, \boldsymbol{X}'_T\right) = \left\| \frac{1}{n_1} \sum_{i=1}^{n_1} \phi\left(x'_{S_i}\right) - \frac{1}{n_2} \sum_{j=1}^{n_2} \phi\left(x'_{T_j}\right) \right\|_{\mathrm{H}}^2 \tag{9}$$

Thus, a sick nonlinear mapping ϕ can be found by minimizing the quantity. However, it is extremely difficult to solve the mapping ϕ and direct optimization of the quantity can stuck ϕ in poor local minima. According to the unsupervised dimensionality reduction method MMDE [70], both the source domain and target domain can be embedded into a public low-dimensional space by learning the kernel matrix K. The kernel mapping can be considered as: $\boldsymbol{X} \to \phi(\boldsymbol{X}) = [\phi(x_1), \phi(x_2), \ldots, \phi(x_n)]$, and $\boldsymbol{K} = \phi(\boldsymbol{X})^{\mathrm{T}}\phi(\boldsymbol{X})$. Specifically, after mapping, $\boldsymbol{X}_S$ from source domain and $\boldsymbol{X}_T$ from target domain can be written as: $\begin{bmatrix} \langle \phi(\boldsymbol{X}_S)\, \phi(\boldsymbol{X}_S) \rangle & \langle \phi(\boldsymbol{X}_S)\, \phi(\boldsymbol{X}_T) \rangle \\ \langle \phi(\boldsymbol{X}_T)\, \phi(\boldsymbol{X}_S) \rangle & \langle \phi(\boldsymbol{X}_T)\, \phi(\boldsymbol{X}_T) \rangle \end{bmatrix}$, thus, $\boldsymbol{K} = \begin{bmatrix} \boldsymbol{K}_{S,S} & \boldsymbol{K}_{S,T} \\ \boldsymbol{K}_{T,S} & \boldsymbol{K}_{T,T} \end{bmatrix}$. In terms of trace operation trick, the distance between samples from source domain and target domain is equivalent to $\mathrm{tr}(\boldsymbol{KM})$, and subject to constraints on $\boldsymbol{K}$. $\boldsymbol{M}$ is MMD matrices as the formula (10).

$$M_{ij} = \begin{cases} \dfrac{1}{n_S n_S}, & x_i,\ x_j \in D_S \\ \dfrac{1}{n_T n_T}, & x_i,\ x_j \in D_T \\ -\dfrac{1}{n_S n_T}, & \text{otherwise} \end{cases} \tag{10}$$

3.4.2 Local structure preservation

However, reducing the difference in the marginal distributions may destroy the relationship structure between samples, leading to the loss of useful information. Therefore, the affinity matrix here is to preserve the neighborhood structure. First, revisit a dimensionality reduction method called LPP [71]. With the manifold assumption, LPP aims to preserve optimally the neighborhood structure of data. The objective function of LPP can be formulated as

$$\sum_{i,j} (y_i - y_j)^2 S_{ij} \tag{11}$$

where $Y = [y_1, y_2, \ldots, y_n]^{\mathrm{T}}$ is the map of $X = [x_1, x_2, \ldots, x_n]^{\mathrm{T}}$, S is the affinity matrix, calculated in the following two manners [72].

a. Simple-minded:

$$S_{ij} = \begin{cases} 1, & \text{if } x_i \in N_{K(x_j)} || x_j \in N_{K(x_i)} \\ 0, & \text{others} \end{cases} \tag{12}$$

b. Heat-kernel:

$$S_{ij} = \begin{cases} e^{-\frac{\left\| x_i - x_j \right\|^2}{t}}, & \text{if } x_i \in N_{K(x_j)} || x_j \in N_{K(x_i)} \\ 0, & \text{others} \end{cases} \tag{13}$$

where t is the kernel parameter. S_{ij} will be assigned a large value if x_i is the neighborhood of x_j. Based on this important idea, to remain the neighbor relationship of

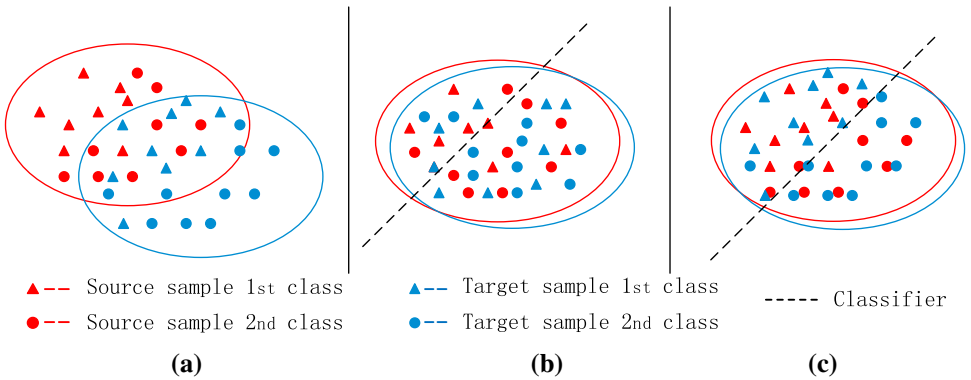

Fig. 1 Illustration of the proposed JLSDA method. **a** The data distribution of the original source domain (training set) and target domain (testing set); **b** the relationship between the samples and the domains only after the alignment of the source and target domain distributions; **c** the data distribution after aligning the distribution of the source domain and the target domain and keeping the neighborhood structure relationship

samples from source and target domain, neighborhood structure preservation of this paper can be defined as

$$\frac{1}{2}\sum_{m,n}\left\|\phi(x'_{S_m})-\phi(x'_{S_n})\right\|_{\mathrm{H}}^2 S_{mn}+\frac{1}{2}\sum_{p,q}\left\|\phi(x'_{T_p})-\phi(x'_{T_q})\right\|_{\mathrm{H}}^2 S_{pq} \tag{14}$$

where $\phi(\boldsymbol{X}'_S)=[\phi(x'_{S_1}),\phi(x'_{S_2}),\ldots,\phi(x'_{S_m})]^{\mathrm{T}}$ is the map of samples from source domain, and $\phi(\boldsymbol{X}'_T)=[\phi(x'_{T_1}),\phi(x'_{T_2}),\ldots,\phi(x'_{T_p})]^{\mathrm{T}}$ is the map of samples from target domain. S_{mn} and S_{pq} are the affinity matrix for source and target domains.

3.4.3 Joint optimization

The proposed JLSDA pursues aligning the distribution of the source and target domains and preserving neighborhood structure. The former reduces the distribution differences between the source domain and the target domain from a large range so that the classifier can match the data better. The latter retains the structure between samples in each domain from a local range, making the original effective information not affected. Therefore, the distribution of alignment is combined with local structure preservation which is important for the small size of PD speech data. Additionally, manifold regularization is also used for local similarity preservation. The main idea of the model is shown in Fig. 1 to reduce distribution differences, while preserving the structure of each domain which may be conducive to classifier classification. Besides the (8), the joint local structure distribution alignment term can be shown as follows.

$$\begin{aligned}\min\ \mathrm{tr}\Bigg(&\frac{1}{n_S^2}\phi(\boldsymbol{X}_S)\boldsymbol{I}\boldsymbol{I}^{\mathrm{T}}\phi^{\mathrm{T}}(\boldsymbol{X}_S)+\frac{1}{n_T^2}\phi(\boldsymbol{X}_T)\boldsymbol{I}\boldsymbol{I}^{\mathrm{T}}\phi^{\mathrm{T}}(\boldsymbol{X}_T)\\&-\frac{1}{n_S n_T}\phi(\boldsymbol{X}_S)\boldsymbol{I}\boldsymbol{I}^{\mathrm{T}}\phi^{\mathrm{T}}(\boldsymbol{X}_T)-\frac{1}{n_S n_T}\phi^{\mathrm{T}}(\boldsymbol{X}_S)\boldsymbol{I}^{\mathrm{T}}\boldsymbol{I}\phi(\boldsymbol{X}_T)\Bigg)\\&+\frac{1}{2}\sum_{m,n}\mathrm{tr}(\phi(x_{S_m})\phi^{\mathrm{T}}(x_{S_m})+\phi(x_{S_n})\phi^{\mathrm{T}}(x_{S_n})\\&-\phi(x_{S_m})\phi^{\mathrm{T}}(x_{S_n})-\phi(x_{S_n})\phi^{\mathrm{T}}(x_{S_m}))S_{mn}\\&+\frac{1}{2}\sum_{p,q}\mathrm{tr}(\phi(x_{T_p})\phi^{\mathrm{T}}(x_{T_p})+\phi(x_{T_q})\phi^{\mathrm{T}}(x_{T_q})\\&-\phi(x_{T_p})\phi^{\mathrm{T}}(x_{T_p})-\phi(x_{T_q})\phi^{\mathrm{T}}(x_{T_p}))S_{pq}\end{aligned} \tag{15}$$

According to the properties of matrix trace and our previous definition, the formula (15) can be simplified into the following formula.

$$\min\mathrm{tr}(\hat{\boldsymbol{K}}\hat{\boldsymbol{M}})+\mathrm{tr}(\phi(\boldsymbol{X}'_S)\phi^{\mathrm{T}}(\boldsymbol{X}'_S)\hat{\boldsymbol{L}}_S)+\mathrm{tr}(\phi(\boldsymbol{X}'_T)\phi^{\mathrm{T}}(\boldsymbol{X}'_T)\hat{\boldsymbol{L}}_T) \tag{16}$$

It is clear that the first part of (15) is similar to the result that discussed in Sect. 2.4.1. where $\hat{\boldsymbol{K}}$ is the kernel matrix and $\hat{\boldsymbol{M}}$ is the MMD matrices, both are obtained from samples after first transfer step. $\phi(\boldsymbol{X}'_S)\phi^{\mathrm{T}}(\boldsymbol{X}'_S)$ is $\hat{\boldsymbol{K}}_S$ and $\phi(\boldsymbol{X}'_T)\phi^{\mathrm{T}}(\boldsymbol{X}'_T)$ is $\hat{\boldsymbol{K}}_T$.$\hat{\boldsymbol{L}}_S=\hat{\boldsymbol{D}}_S-\hat{\boldsymbol{S}}_S$ and $\hat{\boldsymbol{L}}_T=\hat{\boldsymbol{D}}_T-\hat{\boldsymbol{S}}_T$ are the Laplacian matrixes of source domain and target domain, $\hat{\boldsymbol{D}}_{mm}=\sum_n\hat{\boldsymbol{S}}_{mn}$ and $\hat{\boldsymbol{D}}_{qq}=\sum_p\hat{\boldsymbol{S}}_{pq}$ are both diagonal matrixes, $\hat{\boldsymbol{S}}$ is affinity matrix. For convenience, (16) can be simplified to formula (17)

$$\min\mathrm{tr}(\hat{\boldsymbol{K}}\hat{\boldsymbol{M}})+\mathrm{tr}(\hat{\boldsymbol{K}}\cdot\hat{\boldsymbol{L}}) \tag{17}$$

$(\cdot)$ denotes dot multiplication of $\hat{\boldsymbol{K}}$ and $\hat{\boldsymbol{L}}$, $\hat{\boldsymbol{L}}=\begin{bmatrix}\hat{\boldsymbol{L}}_S & \boldsymbol{0}\\ \boldsymbol{0} & \hat{\boldsymbol{L}}_T\end{bmatrix}\in\Re^{N\times N}$.

Due to high computational cost of MMDE, a unified kernel learning method is adopted which utilizes an explicit low-rank representation [67, 73]. Hence, formula (18) can be acquired.

$$\begin{aligned}&\min_{\boldsymbol{W}}\mathrm{tr}(\boldsymbol{W}^{\mathrm{T}}\hat{\boldsymbol{K}}\hat{\boldsymbol{M}}\hat{\boldsymbol{K}}\boldsymbol{W})+\mathrm{tr}(\boldsymbol{W}^{\mathrm{T}}\hat{\boldsymbol{L}}^{*}\boldsymbol{W})+\lambda\mathrm{tr}(\boldsymbol{W}^{\mathrm{T}}\boldsymbol{W})\\&\text{s.t.}\ \ \boldsymbol{W}^{\mathrm{T}}\hat{\boldsymbol{K}}\hat{\boldsymbol{D}}\hat{\boldsymbol{K}}\boldsymbol{W}=\boldsymbol{I}_m\end{aligned} \tag{18}$$

In order to make the problem solution unique, constrain is introduced. $\boldsymbol{W}$ is the mapping matrix, $\hat{\boldsymbol{L}}^{*}$ is dot multiplication of $\hat{\boldsymbol{K}}$ and $\hat{\boldsymbol{L}}$, $\lambda\mathrm{tr}(\boldsymbol{W}^{\mathrm{T}}\boldsymbol{W})$ is the regularization term. On the basis of Lagrange multipliers method, problem (18) can be reformulated as

$$\mathrm{tr}(\boldsymbol{W}^{\mathrm{T}}(\hat{\boldsymbol{K}}\hat{\boldsymbol{M}}\hat{\boldsymbol{K}}+\hat{\boldsymbol{L}}^{*}+\lambda\boldsymbol{I})\boldsymbol{W})-\mathrm{tr}((\boldsymbol{W}^{\mathrm{T}}\hat{\boldsymbol{K}}\hat{\boldsymbol{H}}\hat{\boldsymbol{K}}\boldsymbol{W}-\boldsymbol{I})\boldsymbol{Z}) \tag{19}$$

where $\boldsymbol{Z}$ is a diagonal matrix containing Lagrange multiplier, Setting the derivative of (19) w.r.t. $\boldsymbol{W}$ to zero, then

$$(\hat{\boldsymbol{K}}\hat{\boldsymbol{M}}\hat{\boldsymbol{K}}+\hat{\boldsymbol{L}}^{*}+\lambda\boldsymbol{I})\boldsymbol{W}=\hat{\boldsymbol{K}}\hat{\boldsymbol{H}}\hat{\boldsymbol{K}}\boldsymbol{W}\boldsymbol{Z} \tag{20}$$

The $\boldsymbol{W}$ solutions in (20) are the d leading eigenvectors of $(\hat{\boldsymbol{K}}\hat{\boldsymbol{M}}\hat{\boldsymbol{K}}+\hat{\boldsymbol{L}}^{*}+\lambda\boldsymbol{I})^{-1}\hat{\boldsymbol{K}}\hat{\boldsymbol{H}}\hat{\boldsymbol{K}}$, $d\le n_1+n_2$.

The pseudo-code description of the joint local structure distribution alignment algorithm (JLSDA) is shown as follows.

Algorithm 2: Joint Local Structure Distribution Alignment Algorithm (JLSDA)

Input: train set of Sakar data after first step $X_S^{'}$, test set of Sakar data after first step $X_T^{'}$,initialization parameters: regulation parameter, kernel, number of nearest neighbors, affinity matrix mode

Output: mapping matrix $\boldsymbol{W}$

Procedure:

1: Calculate MMD matrices $\hat{\boldsymbol{M}}$ via (9);

2: Choose proper kernel function, acquiring kernel matrices $\hat{\boldsymbol{K}}$;

3: Constructaffinity matrix $\hat{\boldsymbol{S}}$ of source and target domain respectively via (12) or (13);

4: Construct diagonal matrix $\hat{\boldsymbol{D}}$ of source and target domain respectively, $\hat{\boldsymbol{D}}_{ii} = \sum_j \hat{\boldsymbol{S}}_{ij}$;

5: Construct diagonal matrix $\hat{\boldsymbol{L}}$ of source and target domain respectively, $\hat{\boldsymbol{L}} = \hat{\boldsymbol{D}} - \hat{\boldsymbol{S}}$;

6: Solve $\boldsymbol{W}$ according to formula (20).

4 Experimental results and analysis

This section describes the experiments conducted to test the proposed method's effectiveness for PD diagnosis, mainly including the following experiments: verify the validity of each step of the transfers; explore the impact of import parameters on classification results; compare the representative relevant methods and analyze the computational time.

4.1 Experimental condition

4.1.1 Data

Four speech datasets are adopted for verification: The DARPA TIMIT Acoustic–Phonetic Continuous Speech Corpus (TIMIT), Sakar [15], MaxLittle [3, 74], and DNSH dataset.

The first dataset is used for source domain in the first transfer learning. This standard speech dataset TIMIT contains a total of 6300 sentences, 10 sentences spoken by each of 630 speakers, but there are only 240 samples available for us, including 40 men and 40 women speakers, each one with 3 sentences. The dataset is added with noise (from NOISEX-92 noise dataset) and expansion.

As for PD speech datasets, Little et al. [74] and Sakar et al. [15] provided a speech data set for Parkinson's disease, respectively. The Sakar dataset is the second dataset. There are 40 subjects in Sakar Data, including 20 patients with Parkinson's disease (6 women, 14 men), 20 healthy people (10 women, 10 men). Each subject contains 26 Speech sample segments, and each speech segment contains a variety of pronunciation content, including continuous vowel letter pronunciation, number pronunciation, word pronunciation, and short sentence pronunciation. As for each speech sample, 26-dimensional linear and nonlinear features are extracted to form a feature vector. MaxLittle dataset is the third dataset. The dataset is composed of a range of biomedical voice measurements from 31 people, 23 with Parkinson's disease (PD). For more detailed information on the second and third datasets, please visit the website (https://archive.ics.uci.edu/ml/index.php).

The fourth dataset was collected by the authors and the subjects are collected from the First Affiliated Hospital of the Army Military Medical University, Chongqing, China. The dataset contains recordings of 36 PD patients (16 female (mean ± standard deviation (std): 57.9 ± 9.0) and 20 male (mean ± std: 60.8 ± 10.6)) without receiving treatment (the average and standard deviation age of illness are 7.38 years and 3.58 years, respectively) and 54 PD patients (27 female (mean ± std: 59.7 ± 8.1) and 27 male (mean ± std: 63.2 ± 10.8)) after receiving medication (the average and standard deviation age of illness are 6.82 years and 3.50 years, respectively). Thirteen speech samples were recorded for each person and each speech sample contains 26 features. The recordings were recorded by a microphone (SONY ICD-SX2000) placed at 15 cm away from the participants. The anticipants were asked to read 13 specific characters including '1', '2', '3', '4', '5', '6', '7', '8', '9', '10', 'a', 'o', and 'u'. The speech extraction software was Praat, sampled at 44.1 kHz, with 16-bit resolution.

4.1.2 Experimental criteria

The classification accuracy, sensitivity, and specificity are adopted as the evaluation criteria of experimental results to verify the effectiveness of the proposed algorithm in this paper. The accuracy rate refers to the percentage of the samples that are judged correctly to the total number of samples. Sensitivity and specificity are two commonly used indicators to explain the accuracy of medical diagnostic tests. Since PD speech diagnosis is a binary classification task in this paper, the confusion matrix can be used to

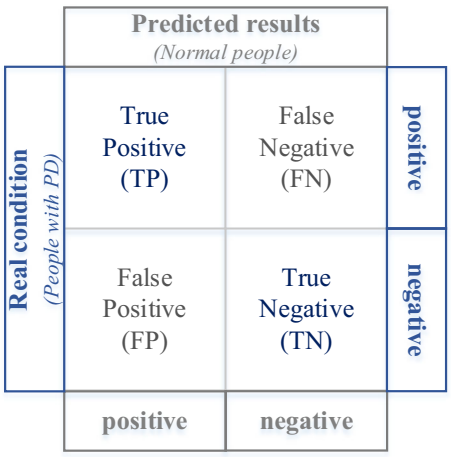

Fig. 2 Confusion matrix for two-class diagnosis of PD

describe the composition of sensitivity and specificity clearly as shown in Fig. 2.

From the confusing matrix, the indicators used in this paper can be expressed as:

$$\text{accuracy (ACC)} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}}$$

$$\text{sensitivity (TPR)} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{specificity (TNR)} = \frac{\text{TN}}{\text{FP} + \text{TN}}$$

The leave-one-subject-out (LOSO) method is applied here, according to the characteristics that multiple samples correspond to one subject in the dataset. This verification method can maximize the number of training samples in small samples case, thus can better reflect the potential of the classification algorithm. Moreover, all samples were sufficiently tested, so the test accuracy was closer to the results in the actual application scenario. Most of the existing algorithms are based on *k*-fold and holdout cross-validation methods, the training samples and test samples are possibly from the same subject, thereby leading to the classification accuracy is not realistic. Different from the two algorithms, the LOSO can guarantee that training samples and test samples are from different subjects, which can ensure that the classification accuracy is not unrealistic and consistent with the actual diagnosis.

4.1.3 Experimental configuration

The experiments use a 64-bit Windows 7 computer and the hardware parameters of the experiment platform are CPU (Intel i3-4170 M), 6 GB memory. The experiments run on Matlab R2018b. The set of parameters in this paper is as follows. In the first step transfer, the random seeds number is 10, the number of main training iterations, feature map iterations, and convolution kernel iterations are 100, 10, and 10, respectively. The number of convolution kernel is from 2 to 8, the size of convolution kernel is 8 * 8. In the second transfer step, the regularization parameter lambda is 0.01, kernel type is 'rbf', the bandwidth for rbf kernel gamma is 100, affinity matrix mode is "simple" mode and the number of nearest neighbors is 1.

4.2 Verification of different steps transfer learning

4.2.1 Performance of first step transfer learning

For convenience, the Sakar dataset is used as the target domain here. In the FT, the main achievement is to transfer the knowledge from TIMIT to Sakar dataset through CSC. The difference between the target domain transformations is shown in Fig. 3. The figure manifests that the information of target domain has increased significantly after transfer learning. Here, whether the information obtained from the source domain contributes to the classification accuracy of the target domain or not is still unknown. By using the Sakar dataset as an example, the data before and after the first step of transfer learning will be handled. The classification algorithms are KNN and SVM. The classification results are shown in Table 1.

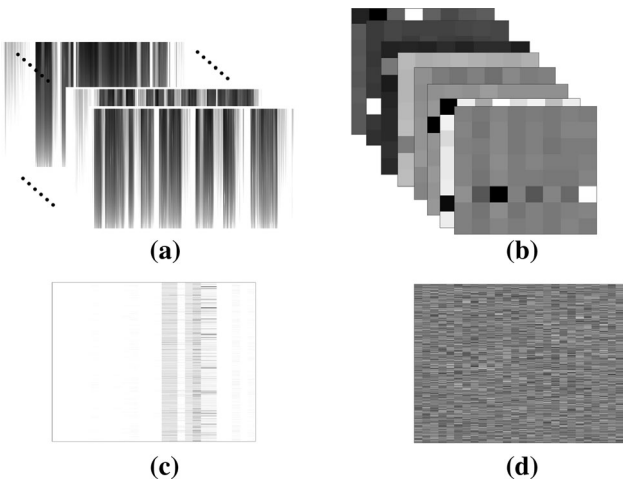

Fig. 3 **a** Sonograms of source domain; **b** feature kernels extracted from source domain; **c** original target domain; **d** target domain after first step transfer

Table 1 First transfer classification accuracy for Sakar dataset

Method	ACC (%)	TPR (%)	TNR (%)
KNN	52.5 (LOSO)	55.0	50.0
SVM (linear)	50.0 (LOSO)	50.0	50.0
FT&KNN	90.0 (LOSO)	85.0	95.0
FT&SVM (linear)	92.5 (LOSO)	95.0	90.0

The results are showed in Table 1. Direct classification accuracy on the Sakar dataset is bad. KNN shows a better result than SVM, with an accuracy of 52.5% and 50.0%, respectively. However, it demonstrates a remarkable improvement in classification accuracy with first step transfer from TIMIT. The accuracy reached 90% for KNN and even achieved 92.5% for SVM. The sensitivity and specificity also are improved for the two classifiers after FT. This result fully illustrates that the information learned from public speech data is conducive to the classification of the target domain that is the first step transfer learning is effective.

4.2.2 Performance of second step transfer learning

In the ST, the JLSDA method is used to diminish the distribution difference between training data and test data, enabling them to keep original local structure. Like the first step transfer experiments, the effectiveness of the method is validated by comparing classification accuracy of untransferred data and transferred data.

Table 2 presents the experimental outcome of ST. Although the experimental results do not improve as significantly as FT, it still worked. The classification accuracies on the KNN classifier and the SVM classifier are increased by 15% and 12.5%, respectively.

4.2.3 Performance of TSTL

The TSTL&KNN means with TSTL with KNN classifier. The TSTL&SVM means TSTL with SVM classifier. The first two experiments proved that every single step of transfer learning is helpful to classification results. In this part, FT is combined with ST algorithms into the two-step sparse transfer learning algorithm. The experimental results are further improved as shown in Table 3. KNN achieved an accuracy of 94.5%, and SVM reached even more about 97.5%, similar to its sensitivity and specificity. The TSTL method has a great effect on the classification accuracy of the final results.

Figure 4 shows that t-SNE visualizations of the effect of the proposed method. Different colors represent samples of different domains. (a), (c), (e) represent the data distribution of the Sakar dataset, MaxLittle dataset, and DNSH dataset before TSTL, and (b), (d), (f) represent the data distribution of the Sakar dataset, MaxLittle dataset, and DNSH dataset after TSTL. It is manifest that data distribution is more compact and even than before TSTL.

Table 2 Second transfer classification accuracy for Sakar dataset

Method	ACC (%)	TPR (%)	TNR (%)
KNN	52.5 (LOSO)	55.0	50.0
SVM (linear)	50.0 (LOSO)	50.0	50.0
ST&KNN	67.5 (LOSO)	65.0	70.0
ST&SVM (linear)	62.5 (LOSO)	80.0	45.0

Table 3 Two-step sparse transfer classification accuracy for Sakar dataset

Method	ACC (%)	TPR (%)	TNR (%)
KNN	52.5 (LOSO)	55.0	50.0
SVM (linear)	50.0 (LOSO)	50.0	50.0
TSTL&KNN	94.5 (LOSO)	94.5	94.5
TSTL&SVM (linear)	97.5 (LOSO)	97.5	97.5

4.2.4 Comparison with unsupervised domain adaptation algorithms

Although there are many studies on Parkinson's classification, there is almost no UDA for Parkinson's speech. Table 4 shows the comparison of the proposed method and four typical UDA methods: two traditional UDA and two deep UDA methods (DAN, DANN). Each method was tested on the Sakar dataset, MaxLittle dataset and DNSH dataset, under LOSO cross-validation.

Compared with other four UDA methods, the ACC of TSTL presents the best results on the three PD datasets. It is noticeable that the effects of the four comparison methods are not ideal, even not reach 50% in the real-world dataset. Moreover, there is no obvious difference in these PD datasets regardless of whether it is deep or non-deep methods. To a certain extent, although these UDA methods have relatively strong versatility, for relatively special datasets such as Parkinson's speech data, to achieve high ACC, it is necessary to design corresponding algorithms according to its data characteristics. Due to the imbalance of positive and negative samples in the MaxLittle, TPR and TNR have a great difference in the four compared methods. However, the proposed method learned speech structure to enhance the generalization ability of the classifier. In general, the proposed algorithm is better than the popular domain adaptation algorithms.

4.3 Effect of parameters on the proposed algorithm's performance

4.3.1 Effect of convolution kernel number

Convolution kernel is one of the main parameters of TSTL, therefore, it is necessary to study its effect on the performance of algorithm. For the Sakar dataset, when the

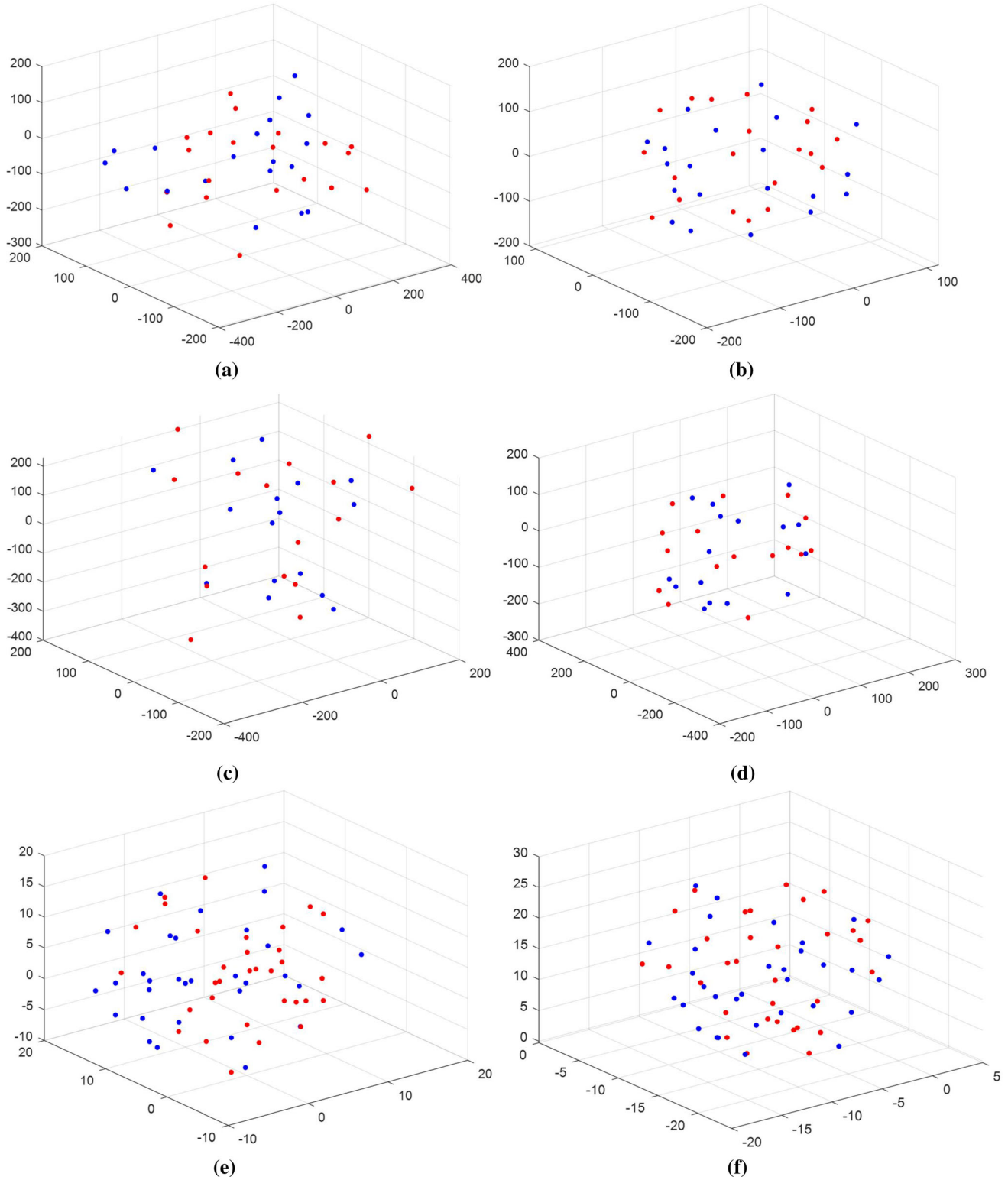

Fig. 4 The t-SNE visualizations of TSTL on three PD speech datasets. **a** Non-TSTL on Sakar; **b** TSTL on Sakar; **c** non-TSTL on MaxLittle; **d** TSTL on MaxLittle; **e** Non-TSTL on DNSH; **f** TSTL on DNSH

convolution kernel number is taken from 2 to 8, after 10 repetitions, the relationship between the number of the convolution kernels and the classification accuracy is shown in Fig. 5. The abscissa represents the number of convolution kernels. With different numbers of convolution kernels, each convolution kernel corresponds to a

Table 4 The comparison of the UDA classification result of the proposed algorithm based on three datasets

Method	Dataset	ACC (%)	TPR (%)	TNR (%)
TCA (LOSO)	Sakar	55.00	65.00	45.00
	MaxLittle	75.00	100.00	0.00
	DNSH	46.88	59.38	34.38
CORAL (LOSO)	Sakar	52.50	50.00	55.00
	MaxLittle	75.00	100.00	0.00
	DNSH	48.44	56.52	40.63
DAN (LOSO)	Sakar	62.50	65.00	60.00
	MaxLittle	66.88	84.17	15.00
	DNSH	45.94	45.66	46.25
DANN (LOSO)	Sakar	54.25	54.50	54.00
	MaxLittle	72.81	93.75	10.00
	DNSH	47.67	54.06	41.25
TSTL (LOSO)	Sakar	**97.50**	**97.50**	**97.50**
	MaxLittle	**96.87**	**100.00**	**87.50**
	DNSH	**90.63**	**90.63**	**90.63**

ACC accuracy; *TPR* true positive rate; *TNR* true negative rate

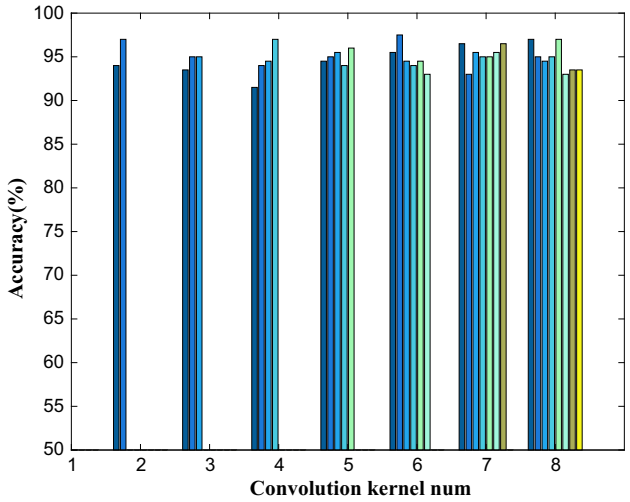

Fig. 5 Relationship between the convolution kernel number and classification accuracy for Sakar dataset

result. The ordinate is the classification accuracy. Each convolution kernel will produce a corresponding result.

All convolution results are more than 90% of the bar graph. Comparing all the results, the classification accuracy rate has a minimum value when the kernel number is 4, and a maximum value when the kernel number is 6. Therefore, the feature kernel leading high classification result can be chosen to obtain a suitable feature map in actual operation. Overall, the results are relatively satisfactory.

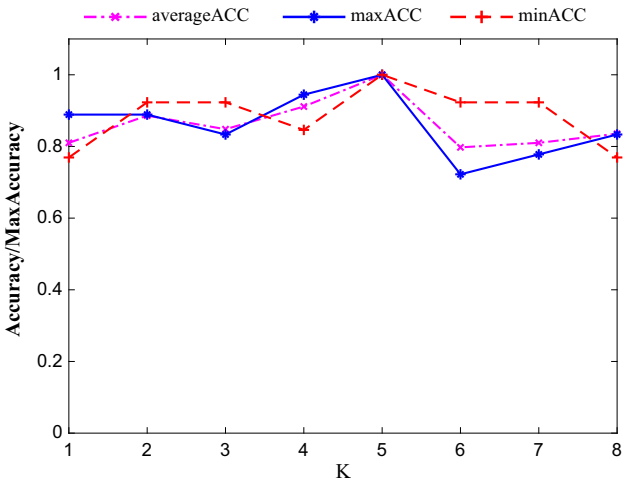

Fig. 6 Relative accuracy of neighbor sample number on Sakar dataset

4.3.2 Effect of neighbor sample number

The nearest neighbor sample number plays a critical role in preserving the neighborhood structure of TSTL. According to the characteristics of Sakar PD speech datasets, nearest neighbor sample number K from 1 to 8 is selected for experiments. Relative accuracy is adopted here to intuitively explore the effect of the number of neighbors on the results. Figure 6 depicts the relative accuracy of maximum accuracy, average accuracy, and minimum accuracy for different numbers of neighbor samples. The results show that average accuracy reaches the maximum value when K is 5. The maximum accuracy and average accuracy is with the same case. The slope linear regression through the data point is adopted to show the relationship between three relative accuracies and neighbor samples. When K is less than 5, the relative accuracy of maximum accuracy, average accuracy, minimum accuracy increase with neighbor samples at the rate of 0.027778, 0.040506, and 0.38462. While K is greater than 5, the relative accuracy of maximum accuracy, average accuracy, minimum accuracy decreases with neighbor samples at the rate of − 0.04444, − 0.0481, − 0.06923, respectively. So it seems that too many or too few neighbor samples will not have a positive effect on classifier classification. Therefore, it is necessary to find a suitable neighbor sample number for classification.

4.4 Comparison with representative PD algorithms

On the Sakar dataset, the comparison results of the proposed algorithm with other representative algorithms are presented in Table 5. Excepting the relevant published algorithms of PD speech diagnosis, the proposed algorithm

Table 5 The comparison of the classification result of the proposed algorithm based on Sakar dataset

Study	Method	ACC (%)	TPR (%)	TNR (%)
Canturk and Karabiber [75]	4 Feature Selection Methods & 6 Classifiers	57.50	54.28	80.00
Eskidere et al. [76]	Random Subspace Classifier Ensemble	74.17	–	–
Zhang et al. [77]	MENN&RF	81.50	92.50	70.50
Benba et al. [78]	HFCC + SVM	87.50	90.00	85.00
Li et al. [79]	Hybrid feature learning&SVM	82.50	85.00	80.00
Vadovsk and Parali [80]	C4.5&C5.0&RF &CART	66.50	–	–
Zhang[81]	LSVM&MSVM &RSVM&CART &KNN&LDA&NB	94.17	50.00	94.92
Benba et al. [82]	MFCC&SVM	82.50	80.00	85.00
Kraipeerapun and Amornsamanku [83]	Stacking&CMTNN	75.00	–	–
Khan et al. [84]	Evolutionary neural network ensembles	90.00	93.00	97.00
Ali et al. [85]	LDA-NN-GA	95.00	95.00	95.00
–	DBN	54.60	52.40	56.80
–	CNN	60.00	63.00	57.00
–	DBN&SVM	50.50	53.00	48.00
–	Autoencoder&SVM	67.50	65.00	70.00
Proposed algorithm	TSTL&SVM	**97.50**	**97.50**	**97.50**

ACC accuracy; *TPR* true positive rate; *TNR* true negative rate

Table 6 The comparison of the classification result of the proposed algorithm based on MaxLittle dataset

Study	Method	ACC (%)	TPR (%)	TNR (%)
Little et al. [74]	Preselection filter + exhaustive search + SVM	91.40	–	–
Shahbaba and Neal [86]	Dirichlet process mixtures	87.70	–	–
Psorakis et al. [87]	mRVMs	89.47	–	–
Guo et al. [88]	GA-EM	93.10	–	–
Sakar and Kursun [27]	Mutual information + SVM	92.75	–	–
Das [89]	ANN decision tree	92.90	–	–
Ozcift and Gulten [90]	Correlation-based feature selection-rotation forest	87.10	–	–
Luukka [91]	Fuzzy entropy measures + similarity	85.03	–	–
Li et al. [92]	Fuzzy-based nonlinear transformation + SVM	93.47	–	–
Spadoto et al. [93]	PSO + OPF harmony search + OPF gravitational search + OPF	84.01	–	–
Polat [94]	FCMFW + KNN	97.93	–	–
Chen et al. [95]	PCA-fuzzy KNN	96.07	–	–
Ali et al. [17]	DBN	94.00	–	–
Åström and Koker [96]	Parallel ANN	91.20	90.50	93.00
Daliri [97]	SVM with Chi-square distance kernel	91.20	91.71	89.92
Zuo et al. [98]	PSO-fuzzy KNN	97.47	98.16	96.57
Kadam and Jadhav [99]	FESA-DNN	93.84	95.23	90.00
Ma et al. [100]	SVM-RFE	96.29	95.00	97.50
Cai et al. [19]	RF-BFO-SVM	97.42	99.29	91.50
Dash et al. [101]	ECFA-SVM	97.95	97.90	–
Gürüler [102]	KMCFW-CVANN	99.52	100.00	99.47
–	SVM (linear kernel)	75.00	100.00	0.00
–	SVM (RBF kernel)	75.00	100.00	0.00
Proposed algorithm	TSTL&SVM	96.87	**100.00**	87.50

ACC accuracy; *TPR* true positive rate; *TNR* true negative rate

Table 7 The comparison of the classification result of the proposed algorithm based on DNSH dataset

Study	Method	ACC (%)	TPR (%)	TNR (%)
–	KNN	52.5 (LOSO)	55.0	50.0
–	SVM (linear kernel)	50.0 (LOSO)	50.0	50.0
Proposed algorithm	TSTL&SVM	90.63 (LOSO)	90.63	90.63

is also compared with the relevant deep learning algorithms, including the DBN, CNN, and deep autoencoder algorithm.

As shown in Table 5, for the Sakar dataset, it is difficult to achieve an excellent result, and only a handful of algorithms can reach an accuracy of 90%. From the methodological point, it is evident that deep learning methods are not better than traditional machine learning methods and most traditional methods have higher accuracy than the former. This also confirmed that the deep learning method is not suitable for the datasets of the small sample such as the Parkinson's speech dataset since it requires a large number of samples to train a good model. Holdout, *K*-fold, and LOSO three different cross-validation methods are used in different algorithms above. But strictly speaking, the LOSO method is more suitable for the evaluation of Parkinson's speech data model, because one subject contains more than one speech sample and the LOSO method can ensure samples of the training set and test set are from the different subject. Not the same as LOSO, the training set and test set of *K*-fold and Holdout may contain samples from the same subject, make the prediction results in the experiment better than the prediction results in the real application scenario. As to the Holdout method, the final evaluation result has a great relationship with the order of the original data. In terms of accuracy, the average accuracy rate of the proposed algorithm (TSTL&SVM) reached 97.5% and achieved better results than other methods.

Table 6 shows the classification and comparison results of this proposed algorithm and the representative algorithms on the MaxLittle dataset. The proposed algorithm is compared with the other representative algorithms on this dataset. Besides, the proposed algorithm is compared with the most relevant algorithms, including the SVM with linear and radial kernels, DBN, CNN and the deep autoencoder algorithm.

Table 8 The time cost of the proposed algorithm on PD speech datasets

Dataset	Sakar	MaxLittle	DNSH
Time cost (s)	25.188	3.269	18.133

As shown in Table 6, the compared methods on the MaxLittle dataset are based on hold-one-out and tenfold. The holdout is more contingent, and even when tenfold is adopted, there is still no deliberate effort to avoid the fact that the training samples and test samples come from the same subject. Therefore, the accuracies of the methods are unreliable since they are perhaps higher than they would be in practice. As Table 6 shows under LOSO, the proposed algorithm achieves 96.87%. Although the accuracy is lower than some comparison algorithms, the accuracy is based on LOSO and reliable since it more reflects the real accuracy.

It can also be found from Table 7 that the proposed algorithm achieves the best results on the DNSH dataset. The SVM and KNN are adopted as popular classifiers. Outperforming the SVM, the average classification accuracy of the proposed algorithm reaches 90.63%, proving that it is quite effective even on the DNSH dataset of Chinese PD patients.

4.5 Analysis of computational time

First, the Table 8 presents the run time of the proposed algorithm on Sakar, MaxLittle, DNSH datasets, respectively. The computation time of the proposed algorithm on the Sakar dataset under different subject size is provided in Fig. 7. The subject size means the size of features * segments. For example, there are 16 features and 16 segments, so the subject size is. The run time includes the total time cost for dealing with the training set and test set. Notably, all the procedures are implemented in the computer of Intel Core i3CPU, 3.7 GHz, and 6 GB RAM.

Seeing from Table 8, the time costs of the proposed algorithm on the three PD datasets are acceptable in practical applications. Seeing from Fig. 6, the computational time and the slope increase as the subject size increases. But the more the subject size is, the better accuracy will be. Therefore, it is necessary to find the suitable subject size for a satisfactory balance. As described above, the apt feature extraction and the coordinate selection of samples and features are needed.

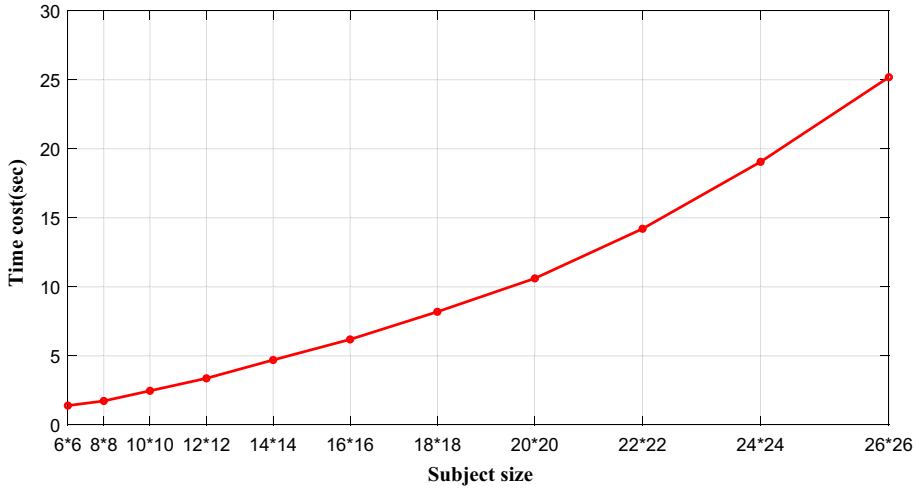

Fig. 7 Time cost of different sample size on Sakar dataset

5 Conclusions

This paper presents unsupervised two-step sparse transfer learning method (TSTL), an efficient approach to replenish information of PD speech samples and reduce the distribution differences between the source domain and target domain. The TSTL method works well on various representative PD speech datasets. Unlike previous PD classification methods, the proposed method used CSC to learn efficient speech structure and designed JLSDA to eliminate discrepancy between training set and test set. The TSTL shows effective results not only on two public datasets but also on the real-world dataset collected by the authors. In the future study, the proposed method will be applied into the PD speech data and motion sensor data together. Besides, the method will be considered for other neurological diseases diagnosis with small sample size.

Acknowledgements We are grateful for the support of the National Natural Science Foundation of China NSFC (No. 61771080); the Fundamental Research Funds for the Central Universities (2019CDQYTX019, 2019CDCGTX306), the Basic and Advanced Research Project in Chongqing (cstc2018jcyjAX0779, cstc2020jcyj-msxmX0523, cstc2020jcyj-msxmX0100); and the Chongqing Technology Innovation and Application Development Project (cstc2020jscx-fyzx0212).

Data availability The data and code can be found in https://share.weiyun.com/14a0OH0B.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflicts of interest related to this work.

References

1. Mirarchi D, Vizza P, Tradigo G et al (2017) Signal analysis for voice evaluation in Parkinson's disease. In: 2017 IEEE International conference on healthcare informatics (ICHI). IEEE, pp 530–535
2. Vollstedt EJ, Kasten M, Klein C et al (2019) Using global team science to identify genetic Parkinson's disease worldwide. Ann Neurol 86(2):153
3. Tsanas A, Little MA, Mcsharry PE et al (2012) Novel speech signal processing algorithms for high-accuracy classification of Parkinson's disease. IEEE Trans Biomed Eng 59(5):1264–1271
4. Gümüşçü A, Karadağ K, Tenekecı ME et al (2017) Genetic algorithm based feature selection on diagnosis of Parkinson disease via vocal analysis. In: 2017 25th Signal processing and communications applications conference (SIU). IEEE, pp 1–4
5. Emrani S, McGuirk A, Xiao W (2017) Prognosis and diagnosis of Parkinson's disease using multi-task learning. In: Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining, pp 1457–1466
6. Tsanas A, Little MA, Mcsharry PE et al (2010) Accurate telemonitoring of Parkinson's disease progression by noninvasive speech tests. IEEE Trans Biomed Eng 57(4):884–893
7. Gillivan-Murphy P, Miller N, Carding P (2019) Voice tremor in Parkinson's disease: an acoustic study. J Voice 33(4):526–535
8. Wroge TJ, Özkanca Y, Demiroglu C et al (2018) Parkinson's disease diagnosis using machine learning and voice. In: 2018 IEEE Signal processing in medicine and biology symposium (SPMB). IEEE, pp 1–7
9. Magee M, Copland D, Vogel AP (2019) Motor speech and non-motor language endophenotypes of Parkinson's disease. Expert Rev Neurother 19:1191–1200
10. Orozco-Arroyave JR, Belalcazar-Bolanos EA, Arias-Londoño JD et al (2015) Characterization methods for the detection of multiple voice disorders: neurological, functional, and laryngeal diseases. IEEE J Biomed Health Inform 19(6):1820–1828
11. Skodda S, Visser W, Schlegel U (2011) Vowel articulation in Parkinson's disease. J Voice 25(4):467–472
12. Orozco-Arroyave JR, Hönig F, Arias-Londoño JD et al (2016) Automatic detection of Parkinson's disease in running speech spoken in three different languages. J Acoust Soc Am 139(1):481–500
13. Kalf J, De Swart B, Bloem BR et al (2007) 3.414 Guidelines for speech–language therapy in Parkinson's disease. Parkinsonism Relat Disord 13(08):S183–S184
14. Zou N, Huang X (2018) Empirical Bayes transfer learning for uncertainty characterization in predicting Parkinson's disease severity. IISE Trans Healthcare Syst Eng 8(3):209–219

15. Sakar BE, Isenkul ME, Sakar CO et al (2013) Collection and analysis of a Parkinson speech dataset with multiple types of sound recordings. IEEE J Biomed Health Inform 17(4):828–834
16. Naseer A, Rani M, Naz S et al (2020) Refining Parkinson's neurological disorder identification through deep transfer learning. Neural Comput Appl 32(3):839–854
17. Al-Fatlawi AH, Jabardi MH, Ling SH (2016) Efficient diagnosis system for Parkinson's disease using deep belief network. In: 2016 IEEE Congress on evolutionary computation (CEC). IEEE, pp 1324–1330
18. Derya A, Akif D (2016) An expert diagnosis system for Parkinson disease based on genetic algorithm-wavelet kernel-extreme learning machine. Parkinson's Dis 2016:1–9
19. Cai Z, Gu J, Chen H et al (2017) A new hybrid intelligent framework for predicting Parkinson's disease. IEEE Access 5:17188–17200
20. Ozkan H (2016) A comparison of classification methods for telediagnosis of Parkinson's disease. Entropy 18(4):115
21. Yang S, Zheng F, Luo X et al (2014) Effective dysphonia detection using feature dimension reduction and kernel density estimation for patients with Parkinson's disease. PLoS ONE 9(2):e88825
22. Shahbakhti M, Taherifar D, Sorouri A (2013) Linear and non-linear speech features for detection of Parkinson's disease. In: The 6th 2013 biomedical engineering international conference. IEEE, pp 1–3
23. Shahbakhi M, Far DT, Tahami E (2014) Speech analysis for diagnosis of Parkinson's disease using genetic algorithm and support vector machine. J Biomed Sci Eng 7(4):147–156
24. Behroozi M, Sami A (2016) A multiple-classifier framework for Parkinson's disease detection based on various vocal tests. Int J Telemed Appl 2016(11, supplement 5):1–9
25. Vásquez-Correa JC, Orozco-Arroyave JR, Arora R et al (2017) Multi-view representation learning via GCCA for multimodal analysis of Parkinson's disease. In: 2017 IEEE International conference on acoustics, speech and signal processing (ICASSP). IEEE, pp 2966–2970
26. Mekyska J, Rektorova I, Smekal Z (2011) Selection of optimal parameters for automatic analysis of speech disorders in Parkinson's disease. In: 2011 34th International conference on telecommunications and signal processing (TSP). IEEE, pp 408–412
27. Sakar CO, Kursun O (2010) Telediagnosis of Parkinson's disease using measurements of dysphonia. J Med Syst 34(4):591–599
28. Su M, Chuang KS (2015) Dynamic feature selection for detecting Parkinson's disease through voice signal. In: 2015 IEEE MTT-S 2015 international microwave workshop series on RF and wireless technologies for biomedical and healthcare applications (IMWS-BIO). IEEE, pp 148–149
29. Caesarendra W, Ariyanto M, Setiawan JD et al (2014) A pattern recognition method for stage classification of Parkinson's disease utilizing voice features. In: 2014 IEEE Conference on biomedical engineering and sciences (IECBES). IEEE, pp 87–92
30. Kaya E, Findik O, Babaoglu I et al (2011) Effect of discretization method on the diagnosis of Parkinson's disease. Int J Innov Comput Inf Control 7:4669–4678
31. Benba A, Jilbab A, Hammouch A (2014) Hybridization of best acoustic cues for detecting persons with Parkinson's disease. In: 2014 Second world conference on complex systems (WCCS). IEEE, pp 622–625
32. Galaz Z, Mekyska J, Mzourek Z et al (2016) Prosodic analysis of neutral, stress-modified and rhymed speech in patients with Parkinson's disease. Comput Methods Prog Biomed 127:301–317
33. Naranjo L, Pérez CJ, Campos-Roca Y et al (2016) Addressing voice recording replications for Parkinson's disease detection. Expert Syst Appl 46:286–292
34. Hirschauer TJ, Adeli H, Buford JA (2015) Computer-aided diagnosis of Parkinson's disease using enhanced probabilistic neural network. J Med Syst 39(11):179
35. Alqahtani EJ, Alshamrani FH, Syed HF et al (2018) Classification of Parkinson's disease using NNge classification algorithm. In: 2018 21st Saudi computer society national computer conference (NCC). IEEE, pp 1–7
36. Pan SJ, Yang Q (2010) A survey on transfer learning. IEEE Trans Knowl Data Eng 22(10):1345–1359
37. Kim DH, Wit H, Thurston M (2018) Artificial intelligence in the diagnosis of Parkinson's disease from ioflupane-123 single-photon emission computed tomography dopamine transporter scans using transfer learning. Nucl Med Commun 39(10):887–893
38. Das D, Lee CSG (2018) Sample-to-sample correspondence for unsupervised domain adaptation. Eng Appl Artif Intell 73(AUG.):80–91
39. Sun B, Feng J, Saenko K (2016) Return of frustratingly easy domain adaptation. In: Thirtieth AAAI conference on artificial intelligence
40. Sun B, Saenko K (2016) Deep coral: correlation alignment for deep domain adaptation. In: European conference on computer vision. Springer, Cham, pp 443–450
41. Sakurai S, Uchiyama H, Shimada A et al (2018) Two-step transfer learning for semantic plant segmentation. In: 7th International conference on pattern recognition applications and methods
42. An G, Yokota H, Motozawa N et al (2019) Deep learning classification models built with two-step transfer learning for age related macular degeneration diagnosis. In: 2019 41st Annual international conference of the IEEE engineering in medicine and biology society (EMBC). IEEE
43. Zhang R, Guo Z, Sun Y et al (2020) COVID19X-rayNet: a two-step transfer learning model for the COVID-19 detecting problem based on a limited number of chest X-ray images. Interdiscip Sci Comput Life Sci 12:1–11
44. Zhang H, Patel VM (2018) Convolutional sparse and low-rank coding-based image decomposition. IEEE Trans Image Process 27(5):2121–2133
45. Hu X, Heide F (2018) Convolutional sparse coding for RGB + NIR imaging. IEEE Trans Image Process 27(4):1611–1625
46. Wohlberg B (2016) Efficient algorithms for convolutional sparse representations. IEEE Trans Image Process 25(1):301–315
47. Hang Chang; Ju Han; Cheng Zhong (2018) Unsupervised transfer learning via multi-scale convolutional sparse coding for biomedical applications. IEEE Trans Pattern Anal Mach Intell 40(5):1182–1194
48. Pan SJ, Tsang IW, Kwok JT et al (2011) Domain adaptation via transfer component analysis. IEEE Trans Neural Netw 22(2):199–210
49. Ganin Y, Lempitsky V (2014) Unsupervised domain adaptation by backpropagation. arXiv preprint arXiv:1409.7495
50. Bousmalis K, Trigeorgis G, Silberman N et al (2016) Domain separation networks. In: Advances in neural information processing systems, NIPS 2016, pp 343–351
51. Kang G, Zheng L, Yan Y et al (2018) Deep adversarial attention alignment for unsupervised domain adaptation: the benefit of target expectation maximization. In: Proceedings of the European conference on computer vision (ECCV), pp 401–416
52. Csurka G (2017) A comprehensive survey on domain adaptation for visual applications. In: Csurka G (ed) Domain adaptation in computer vision applications. Springer, Cham, pp 1–35

53. Pinheiro PO (2018) Unsupervised domain adaptation with similarity learning. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 8004–8013
54. Sener O, Song HO, Saxena A et al (2016) Learning transferrable representations for unsupervised domain adaptation. In: Advances in neural information processing systems, NIPS 2016, pp 2110–2118
55. Haeusser P, Frerix T, Mordvintsev A et al (2017) Associative domain adaptation. In: Proceedings of the IEEE international conference on computer vision, pp 2765–2773
56. Saito K, Ushiku Y, Harada T et al (2017) Adversarial dropout regularization. arXiv preprint arXiv:1711.01575
57. Saito K, Watanabe K, Ushiku Y et al (2018) Maximum classifier discrepancy for unsupervised domain adaptation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3723–3732
58. Pei Z, Cao Z, Long M et al (2018) Multi-adversarial domain adaptation. In: Thirty-second AAAI conference on artificial intelligence
59. Liu F, Zhang G, Lu J (2020) Heterogeneous domain adaptation: an unsupervised approach. IEEE Trans Neural Netw Learn Syst 31:5588–5602
60. Long M, Wang J, Ding G et al (2013) Transfer feature learning with joint distribution adaptation. In: Proceedings of the IEEE international conference on computer vision, pp 2200–2207
61. Gong B, Shi Y, Sha F et al (2015) Geodesic flow kernel for unsupervised domain adaptation. In: 2012 IEEE Conference on computer vision and pattern recognition. IEEE
62. Wang J, Feng W, Chen Y et al (2018) Visual domain adaptation with manifold embedded distribution alignment. In: 2018 ACM International conference on multimedia, pp 402–410
63. Long M, Cao Y et al (2018) Transferable representation learning with deep adaptation networks. IEEE Trans Pattern Anal Mach Intell 41:3071–3085
64. Long M, Zhu H, Wang J et al (2017) Deep transfer learning with joint adaptation networks. In: The 34th international conference on machine learning, Sydney, pp 2208–2217
65. Ganin Y, Ustinova E, Ajakan H et al (2017) Domain-adversarial training of neural networks. J Mach Learn Res 17(1):2096–2030
66. Long M, Cao Z, Wang J et al (2018) Conditional adversarial domain adaptation. In: 32nd Conference on neural information processing systems (NeurIPS 2018), Montreal, Canada pp 1640–1650
67. Kononenko I (1994) Estimating attributes: analysis and extensions of RELIEF. In: European conference on machine learning. Springer, Berlin, pp 171–182
68. Boyd S, Parikh N (2011) Distributed optimization and statistical learning via the alternating direction method of multipliers. Found Trends Mach Learn 3(1):1–122
69. Wang J, Chen Y, Hao S et al (2017) Balanced distribution adaptation for transfer learning. In: 2017 IEEE International conference on data mining (ICDM). IEEE, pp 1129–1134
70. Pan SJ, Kwok JT, Yang Q (2008) Transfer learning via dimensionality reduction. In: AAAI, vol 8, pp 677–682
71. He X, Niyogi P (2004) Locality preserving projections. In: Advances in neural information processing systems, NIPS 2004, pp 153–160
72. Belkin M, Niyogi P (2002) Laplacian eigenmaps and spectral techniques for embedding and clustering. Im: NIPS'01: Proceedings of the 14th international conference on neural information processing systems: natural and synthetic, January 2001, pp 585–591
73. Schölkopf B, Smola A, Müller KR (1998) Nonlinear component analysis as a kernel eigenvalue problem. Neural Comput 10(5):1299–1319
74. Little MA, Mcshappy PE, Roberts SJ et al (2007) Exploiting nonlinear recurrence and fractal scaling properties for voice disorder detection. BioMed Eng OnLine 6:23–41
75. Canturk I, Karabiber F (2016) A machine learning system for the diagnosis of Parkinson's disease from speech signals and its application to multiple speech signal types. Arab J Sci Eng 41(12):5049–5059
76. Eskıdere Ö, Karatutlu A, Ünal C (2015) Detection of Parkinson's disease from vocal features using random subspace classifier ensemble. In: 2015 Twelve international conference on electronics computer and computation (ICECCO). IEEE, pp 1–4
77. Zhang H-H, Yang L, Liu Y, Wang P, Yin J, Li Y, Qiu M, Zhu X, Yan F (2016) Classification of Parkinson's disease utilizing multi-edit nearest-neighbor and ensemble learning algorithms with speech samples. BioMed Eng OnLine 15(1):122–143
78. Benba A, Jilbab A, Hammouch A (2017) Using human factor cepstral coefficient on multiple types of voice recordings for detecting patients with Parkinson's disease. IRBM 38(6):346–351
79. Li Y, Zhang C, Jia Y et al (2017) Simultaneous learning of speech feature and segment for classification of Parkinson disease. In: 2017 IEEE 19th International conference on e-health networking, applications and services (Healthcom). IEEE, pp 1–6
80. Vadovský M, Paralič J (2017) Parkinson's disease patients classification based on the speech signals. In: 2017 IEEE 15th International symposium on applied machine intelligence and informatics (SAMI). IEEE, pp 000321–000326
81. Zhang YN (2017) Can a smartphone diagnose Parkinson disease? A deep neural network method and telediagnosis system implementation. Parkinson's Dis 2017:1–11
82. Benba A, Jilbab A, Hammouch A (2016) Analysis of multiple types of voice recordings in cepstral domain using MFCC for discriminating between patients with Parkinsons disease and healthy people. Int J Speech Technol 19(3):449–456
83. Kraipeerapun P, Amornsamankul S (2015) Using stacked generalization and complementary neural networks to predict Parkinson's disease. In: 2015 11th International conference on natural computation (ICNC). IEEE, pp 1290–1294
84. Khan MM, Mendes A, Chalup SK (2018) Evolutionary wavelet neural network ensembles for breast cancer and Parkinson's disease prediction. PLoS ONE 13(2):e0192192
85. Ali L, Zhu C, Zhang Z et al (2019) Automated detection of Parkinson's disease based on multiple types of sustained phonations using linear discriminant analysis and genetically optimized neural network. IEEE J Transl Eng Health Med 7:1–10
86. Shahbaba B, Neal R (2009) Nonlinear models using Dirichlet process mixtures. J Mach Learn Res 10(12):1829–1850
87. Psorakis I, Damoulas T, Girolami MA (2010) Multiclass relevance vector machines: sparsity and accuracy. IEEE Trans Neural Netw 21(10):1588–1598
88. Guo PF, Bhattacharya P, Kharma NN (2010) Advances in detecting Parkinson's disease. In: Medical biometrics, second international conference, ICMB, Hong Kong, China, June. DBLP
89. Das R (2010) A comparison of multiple classification methods for diagnosis of Parkinson disease. Expert Syst Appl 37(2):1568–1572
90. Ozcift A, Gulten A (2011) Classifier ensemble construction with rotation forest to improve medical diagnosis performance of machine learning algorithms. Comput Methods Prog Biomed 104(3):443–451
91. Luukka P (2011) Feature selection using fuzzy entropy measures with similarity classifier. Expert Syst Appl 38(4):4600–4607

92. Li DC, Liu CW, Hu SC (2011) A fuzzy-based data transformation for feature extraction to increase classification performance with small medical data sets. Artif Intell Med 52(1):45–52
93. Spadoto AA, Guido RC, Carnevali FL et al (2011) Improving Parkinson's disease identification through evolutionary-based feature selection. In: 2011 Annual international conference of the IEEE engineering in medicine and biology society. IEEE, pp 7857–7860
94. Polat K (2012) Classification of Parkinson's disease using feature weighting method on the basis of fuzzy C-means clustering. Int J Syst Sci 43(4):597–609
95. Chen HL, Huang CC, Yu XG et al (2013) An efficient diagnosis system for detection of Parkinson's disease using fuzzy *k*-nearest neighbor approach. Expert Syst Appl 40(1):263–271
96. Åström F, Koker R (2011) A parallel neural network approach to prediction of Parkinson's disease. Expert Syst Appl 38(10):12470–12474
97. Daliri MR (2013) Chi-square distance kernel of the gaits for the diagnosis of Parkinson's disease. Biomed Signal Process Control 8(1):66–70
98. Zuo WL, Wang ZY, Liu T et al (2013) Effective detection of Parkinson's disease using an adaptive fuzzy *k*-nearest neighbor approach. Biomed Signal Process Control 8(4):364–373
99. Kadam VJ, Jadhav SM (2019) Feature ensemble learning based on sparse autoencoders for diagnosis of Parkinson's disease. In: Kadam V, Jadhav SM (eds) Computing, communication and signal processing. Springer, Singapore, pp 567–581
100. Ma H, Tan T, Zhou H et al (2016) Support vector machine-recursive feature elimination for the diagnosis of Parkinson disease based on speech analysis. In: 2016 Seventh international conference on intelligent control and information processing (ICICIP). IEEE, pp 34–40
101. Dash S, Thulasiram R, Thulasiraman P (2017) An enhanced chaos-based firefly model for Parkinson's disease diagnosis and classification. In: 2017 International conference on information technology (ICIT). IEEE, pp 159–164
102. Gürüler H (2017) A novel diagnosis system for Parkinson's disease using complex-valued artificial neural network with *k*-means clustering feature weighting method. Neural Comput Appl 28(7):1657–1666