Semi-automatic data annotation based on feature-space projection and local quality metrics: An application to cerebral emboli characterization

Yamil Vindas, Blaise Kévin Guépié, Marilys Almar, Emmanuel Roux, Philippe Delachartre

Semi-automatic data annotation based on feature-space projection and local quality metrics: An application to cerebral emboli characterization

Yamil Vindas[a,*], Blaise Kévin Guépié[b], Marilys Almar[c], Emmanuel Roux[a], Philippe Delachartre[a]

[a] Univ Lyon, INSA-Lyon, Université Claude Bernard Lyon 1, UJM-Saint Etienne, CNRS, Inserm, CREATIS UMR 5220, U1294, LYON, F-69100, France
[b] Université de Technologie de Troyes / Laboratoire Informatique et Société Numérique, 10004 Troyes, France
[c] Atys Medical, 17 Parc Arbora, Soucieu-en-Jarrest 69510, France

A B S T R A C T

We propose a semi-supervised learning approach to annotate a dataset with reduced requirements for manual annotation and with controlled annotation error. The method is based on feature-space projection and label propagation using local quality metrics. First, an auto-encoder extracts the features of the samples in an unsupervised manner. Then, the extracted features are projected by a *t*-distributed stochastic neighbor embedding algorithm into a two-dimensional (2D) space. A selection of the best 2D projection is introduced based on the silhouette score. The expert annotator uses the obtained 2D representation to manually label samples. Finally, the labels of the labeled samples are propagated to the unlabeled samples using a K-nearest neighbor strategy and local quality metrics. We compare our method against semi-supervised optimum-path forest and K-nearest neighbor label propagation (without considering local quality metrics). Our method achieves state-of-the-art results on three different datasets by labeling more than 96% of the samples with an annotation error from 7% to 17%. Additionally, our method allows to control the trade-off between annotation error and number of labeled samples. Moreover, we combine our method with robust loss functions to compensate for the label noise introduced by automatic label propagation. Our method allows to achieve similar, and even better, classification performances compared to those obtained using a fully manually labeled dataset, with up to 6% in terms of classification accuracy.

1. Introduction

According to the World Health Organization, stroke is one of the leading causes of disability worldwide (Johnson et al., 2016), and cerebral emboli have been related to the risk of stroke (Wallace et al., 2015). Cerebral emboli can be generated by several medical procedures, such as transcatheter aortic valve implantation (Aggarwal et al., 2018), cerebral angiography (Markus et al., 1993), and patent foramen ovale tests (Serena et al., 2010), and they can occur as a result of a variety of conditions, such as carotid artery stenosis (Rosenkranz et al., 2006).

Several techniques can be used to detect emboli (Wallace et al., 2015), such as magnetic resonance imaging and computed tomography. The main drawbacks of these techniques are that they are invasive and expensive, and they do not allow long-duration monitoring of the cerebral blood flow. A well-suited alternative to solve these drawbacks is transcranial Doppler (TCD) monitoring. TCD monitoring is a non-invasive and relatively cheap ultrasound technique that allows the cerebral blood flow to be monitored over long periods of time (from a few minutes, to a few hours). During the monitoring, high-intensity transient signals (HITS), corresponding to emboli or artifacts, can be detected. In this paper we work with TCD data that were acquired with a portable robotic probe worn by the patient, allowing them free movement and long-duration monitoring without loss of signal.

Moreover, the HITS can be used to discriminate between emboli (solid or gaseous) and artifacts (see Fig. 1). Many studies have tried to detect emboli using TCD data through classical signal-processing techniques such as Fourier transforms and wavelet transforms (Markus and Punter, 2005; Gencer et al., 2013; Serbes and Aydin, 2014; Karahoca and Tunga, 2015; Sombune et al., 2016), and also through machine-learning techniques such as support vector machine (SVM) algorithms (Guépié et al., 2017; Guepie et al., 2019)

* Corresponding author.
E-mail address: yamil.vindas@creatis.insa-lyon.fr (Y. Vindas).

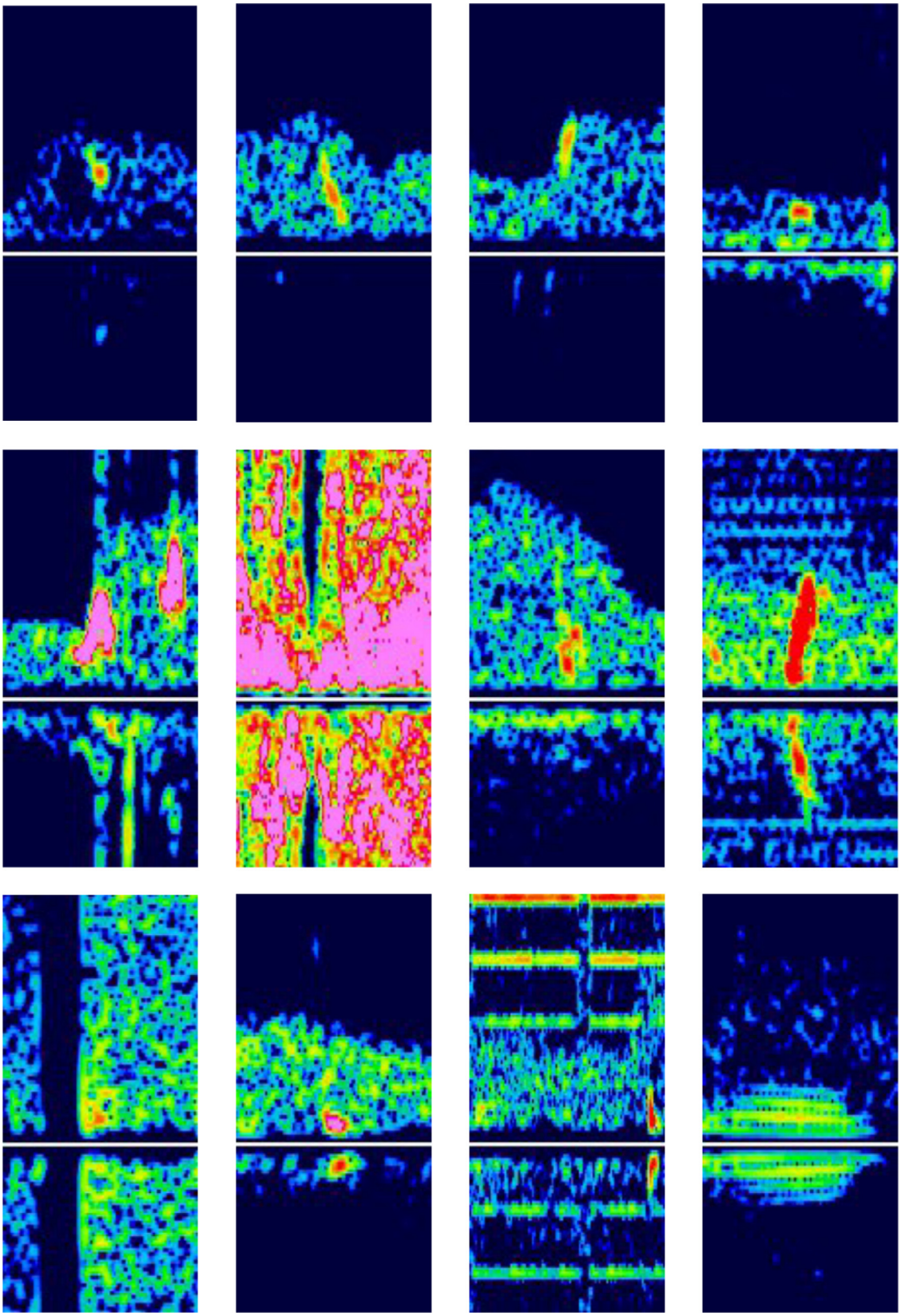


Fig. 1. Examples of spectrograms of high-intensity transient signals. Top row: solid emboli; middle row: gaseous emboli; bottom row: artifacts. Solid emboli usually have lower intensities and shorter durations than gaseous emboli, which are usually 'v'-shaped with higher intensities than solid emboli (they have higher energies). Artifacts are usually symmetric.

and deep-learning techniques such as convolutional neural networks (CNNs) (Sombune et al., 2017; Tafsast et al., 2018). These studies have shown impressive results in cerebral emboli detection and its discrimination from artifacts, although only Guépié et al. (2017); Guepie et al. (2019) used portable TCD data. This is a key point, because portable TCD monitors are more prone to artifacts, making the detection and discrimination task more complex than with conventional TCD data.

Furthermore, one of the main difficulties in real-world deep learning applications is that data acquisition and annotation is costly. More specifically, data annotation can be expensive, time-consuming and often requires expert knowledge (this is particularly true in the medical field). Some semi-supervised learning approaches have tried to address this problem using label propagation (Zhu and Ghahramani, 2002; Benato et al., 2018; 2021), generative models (Kingma et al., 2014), and self-training (Rosenberg et al., 2005). Only a few reports in the literature have applied these methods to TCD data (Vindas et al., 2021). Nevertheless, these approaches work in a high-dimensional space or project the data into a lower-dimensional space, without taking into account the local quality of the projected points to guide the label propagation. Additionally, even if automatic data annotation methods introduce some errors in the labels, not many studies have taken this into account when they have trained their models.

In this study, we propose a framework for semi-automatic data annotation (label propagation) and classification using semi-automatically labeled datasets and we evaluated it on three tasks: emboli classification, organ classification and digit classification. Furthermore, our method relies on three assumptions (structure assumption (Chapelle et al., 2009), the preservation of the local structure during projection and the annotation space coverage) and combines: (1) representation learning and dimensionality reduction techniques; (2) projection quality evaluation; and (3) noise-tolerant loss functions. These concepts are detailed in the following sections.

1.1. Related work

In this subsection, we present four domains that are related to our work: representation learning and dimensionality reduction; semi-supervised learning; noisy-labels learning; and semi-automatic data annotation.

1.1.1. Representation learning, dimensionality reduction, and their qualities

Auto-encoders Tschannen et al. (2018) have been widely used in several problems to extract features from high-dimensional data (Doersch et al., 2015; Chen et al., 2017), and even for anomaly detection (Zhou and Paffenroth, 2017). Representation learning by deep neural networks is close to human perception (Zhang et al., 2018); however, it remains very high dimensional. A commonly used technique is to project the learned representations into a lower dimensional space that can be visualized using some dimensionality reduction techniques (Packer et al., 2013), such as *t*-distributed stochastic neighbor embedding (t-SNE) (Maaten and Hinton, 2008), principal component analysis (PCA) (Jolliffe and Cadima, 2016), isometric feature mapping (ISOMAP) (Tenenbaum, 2000), and uniform manifold approximation and projection (UMAP) (McInnes et al., 2020). Moreover, it has been shown that working on lower dimensional spaces facilitates manual annotation and label propagation (Benato et al., 2018; 2021). Thus, these lower dimensional spaces can then be used for interactive and semi-automatic data annotation (Benato et al., 2018; 2021; Vindas et al., 2021). Furthermore, in the context of data annotation, to reduce annotation errors and have reliable projections, it is important to be able to evaluate the quality of the final projection using global and local projection quality evaluation metrics (Lueks et al., 2011).

1.1.2. Semi-supervised learning and label propagation

Data annotation can be a time consuming and expensive task, particularly for medical applications. This can lead to partially annotated datasets that can be difficult to handle. Semi-supervised learning methods are good candidates to exploit both labeled and unlabeled data. Indeed, while it is possible to use only the labeled data, the rationale behind semi-supervised learning is that unlabeled data can bring important information to the developed models. Different methods have been proposed to propagate the labels from labeled samples to unlabeled samples (these methods often make structure assumptions[1] (Chapelle et al., 2009)): from label propagation (Zhu and Ghahramani, 2002), to generative models (Kingma et al., 2014) and self-training (Rosenberg et al., 2005). Label propagation Zhu and Ghahramani (2002) gives the same label to close samples using a K-nearest neighbor (KNN) strategy; generative models (Kingma et al., 2014) treat the labels of the unlabeled samples as latent variables that can be generated using a learned distribution; and self-training (Rosenberg et al., 2005) trains a model several times with a dataset that is continuously improved and annotated over the iterations through the trained model of the previous iteration. A lot of these methods use machine-learning and deep-learning algorithms, plus some embedding representations as regularization to exploit both the labeled and unlabeled samples (*e.g.*, Laplacian SVMs (Belkin et al., 2006; Sindhwani et al., 2005), deep semi-supervised embedding (Weston et al., 2008)). Other methods, such as optimum-path forest semi-supervised (OPF-semi) (Amorim et al., 2014), propagate labels using a graph structure: the training set (which is composed of labeled and unlabeled samples) is transformed into a graph, then representers of the different classes are computed, and finally, the unlabeled samples are annotated by assigning to them the label of their closest labeled representer. However, to the best of our knowledge, none of these methods take into account the quality of the learned embedding.

1.1.3. Noisy-labels learning and noise-tolerant loss functions

Another aspect encountered with semi-automatic annotation (and more generally, with the annotation of a lot of unlabeled data) is that some errors (or noise) are introduced on the labels. Therefore, we have to find strategies to compensate for this noise that is added to the labels. Several methods allow this problem to be tackled (Song et al., 2021). Robust loss function methods use loss functions that are noise tolerant, such as generalized cross entropy (GCE) (Zhang and Sabuncu, 2018) and symmetric CE (Wang et al., 2019). A similar family of methods known as loss adjustment can lower the negative influence of noisy labeled samples by adjusting them before updating the weights of the model (Song et al., 2019). Robust architecture methods estimate the label-transition matrix, using noise adaptation layers (Goldberger and Ben-Reuven, 2017) or dedicated architectures (Xiao et al., 2015). Robust regularization uses regularization to improve the generalization capability of a model trained on noisy-label data (Pereyra et al., 2017). Finally, other families of methods can be used to select the correctly labelled samples (Song et al., 2019) or to use a set of weak models to re-build the labels of the samples (Yan et al., 2016).

[1] Also known as cluster or manifold assumptions, which say that samples that are in the same structure (*i.e.*, manifold or cluster) are likely to have the same labels.

1.1.4. Semi-automatic data annotation

Deep neural networks have been shown to be efficient in learning representations that can be close to human perception (Zhang et al., 2018). This can justify why some semi-automatic annotation methods start by extracting features in an unsupervised manner using auto-encoders, before projecting the obtained features into a 2D space for manual and automatic annotation (Benato et al., 2018; 2021). To improve the classification performances of models on unseen data, Benato et al. (Benato et al., 2018) proposed the combination of semi-supervised learning with interactive guided manual annotation on a 2D feature space, as images from the modified dataset from the National Institute of Standards and Technology (MNIST) and from microscopic images. Their pipeline relies on an auto-encoder to extract the features, t-SNE to project the data in the 2D space that is used to interactively/ manually label samples, and a Laplacian SVM and OPF to automatically propagate labels from the labeled samples to the unlabeled samples. The same group Benato et al. (2021) improved this pipeline by incorporating the concept of 'confidence' of the classifiers when carrying out the automatic label propagation. This allows the manual annotators to focus just on the difficult samples that classifiers cannot correctly predict with high confidence. Moreover, these two last methods are the closest to our proposed method, along with label propagation (Zhu and Ghahramani, 2002; Vindas et al., 2021), which are based on a KNN strategy. To our knowledge, these methods do not propose to take into account the quality of the 2D projection to do label propagation, nor a strategy to select the optimal projection, nor a strategy to compensate the noise in the labels introduced by automatic annotation.

1.2. Contribution

As discussed above, the combination of semi-automatic annotation (based on representation learning), dimensionality reduction, and label propagation is a promising solution to overcome the difficulty of data annotation. Indeed, representation learning allows to extract features from raw data in an unsupervised manner; dimensionality reduction allows to get lower dimensional spaces easier for the expert to interact with; and label propagation allows to take advantage of the few labeled samples to automatically annotate some unlabeled samples. To do this, we start by extracting features from the data using a deep convolutional auto-encoder (Tschannen et al., 2018; Chen et al., 2017), and then project these features onto a 2D space using dimensionality reduction techniques (*i.e.*, t-SNE (Maaten and Hinton, 2008)). We then select the best projection using the silhouette score metric (Rousseeuw, 1987), and finally propagate the labels based on both global and local quality measures of the projection (Lueks et al., 2011) and a KNN strategy. Furthermore, we use the obtained dataset, which is composed of the original labeled samples and the new labeled samples (with our label propagation method), to train a deep CNN (DCNN) to do classification using a robust loss function that allows compensation for the noise introduced in the labels by our semi-automatic data-annotation method. Thereby, our proposed method is general and composed of flexible blocks which can adapt to different types of dataset (as shown experimentally in Section 3). More specifically, the core elements of our contribution (optimal projection selection and label propagation steps) are generic, automatic and only depend on the feature space obtained from the raw data. Moreover, thanks to the hyperparameters of our propagation method, the user is able to control the trade-off between annotation error and proportion of labeled samples.

To summarize, the main contributions of this paper are as follows:

- We propose a novel methodology for semi-automatic data annotation based on global and local quality metrics with controlled annotation error;
- We introduce a selection strategy to select the best projection for data annotation (obtained using a dimensionality reduction technique);
- We propose to use robust loss functions to improve the classification performances of a classifier trained on a noisy semi-automatic labeled dataset obtained by a semi-automatic annotation method.

The rest of the paper is structured as follows. In Section 2, the semi-automatic data annotation method is presented in detail. In Section 3, the data and experimental evaluation are presented. In Section 4, we discuss the results of the different experiments, and in Section 5, we conclude and give some guidelines to our future work.

2. Proposed method: Semi-supervised data annotation and classification

Let us begin by specifying the three assumptions on which our method is based: the **structure assumption** (Chapelle et al., 2009); the **preservation of the local structure** during projection; and the **annotation space coverage**. The first assumption establishes that samples belonging to the same structure are likely to be part of the same class. The second assumption says that if samples are projected from a high-dimensional space to a lower-dimensional space, their neighborhood should be preserved (even if some errors can be tolerated). Finally, the third assumption means that the few available labeled samples should cover as much as possible the whole annotation space.

Let us assume that we have a dataset $\mathcal{D}$ composed of a large number of unlabeled samples $\mathcal{U}$ ($|\mathcal{U}| = U$, where $|\mathcal{U}|$ is the cardinal of $\mathcal{U}$), and a small number of labeled samples $\mathcal{L}$ ($|\mathcal{L}| = L$) with N classes. Our method (Fig. 2) combines the different approaches that are presented in Section 1.1, and is composed of four steps:

- **Feature extraction**: We start by extracting features in an unsupervised manner using an auto-encoder adapted to our data. Using unsupervised learning techniques allows handcrafted features to be avoided and allows to use all of the available samples from $\mathcal{D}$.
- **Dimensionality reduction**: We reduce the dimension of the latent space of the previous step to obtain a 2D space. This allows more efficient automatic and manual labeling of the samples, as shown in Benato et al. (2021). In this step, we compute different projections and we select the optimal projection using the silhouette score.
- **Automatic label propagation**: By considering the local projection quality of each sample in the 2D space, we propagate the labels of high-quality labeled samples to high-quality unlabeled samples. This allows the creation of a richer training set (*i.e.*, it increases the size of $\mathcal{L}$) with reduced effort.
- **Classification with noisy labels**: Finally, classification is carried out using noisy-label techniques to compensate for the noise introduced by the automatic label propagation.

2.1. Feature extraction

To extract data-specific features from the input samples, we use an auto-encoder, an unsupervised way to obtain a compressed representation of data. It is composed of two parts: an encoder that encodes the information into a latent feature space (in our case, with dimension $>> 2$), and a decoder that uses the extracted features of the input to reconstruct it. Although the principle of our method is generic and can be used for multiple types of data (*e.g.*,

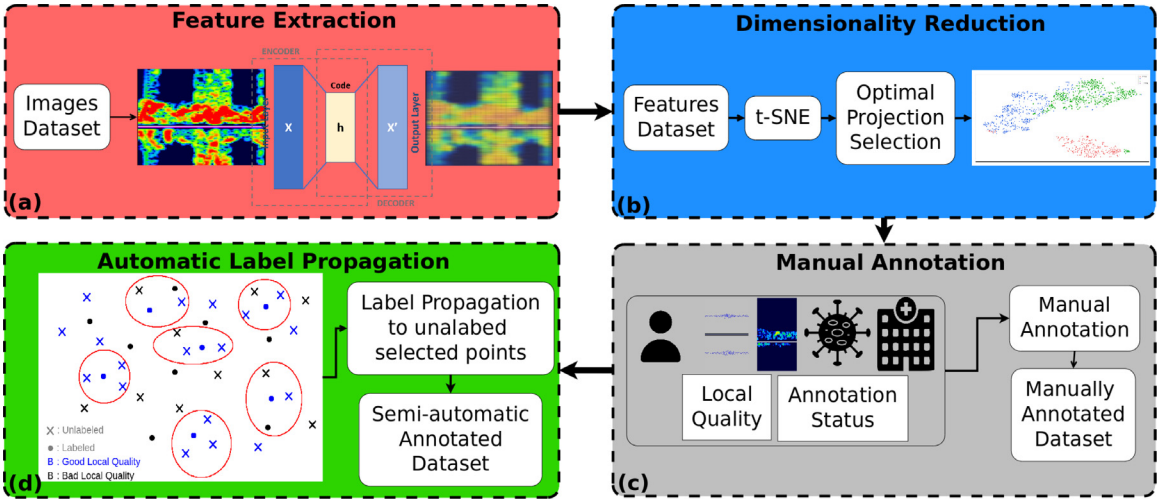


Fig. 2. Proposed semi-supervised annotation pipeline. Our label propagation method (LQ-KNN) is composed of four parts: (a) Feature extraction, where a latent feature space is learned from the input samples; (b) Dimensionality reduction, where the latent feature space of the previous step is projected into a 2D space; (c) Manual annotation, where an expert uses the 2D space obtained with the previous step together with some sample metadata to annotate some samples; (d) Automatic label propagation with a KNN strategy and local quality measures using the previously obtained manually labeled dataset.

time-series, audio, volumes), as we work with images, we use a convolutional auto-encoder to reduce the number of parameters of the model, and to exploit the spatial context of the images.

Moreover, as our objective is to annotate data, the convolutional auto-encoder is trained on the labeled and unlabeled data, which allows the use of all of the available data to improve the learning process.

2.2. Dimensionality reduction

Although the previous step provides considerable reduction of the dimensionality of our problem, it still remains too large for our main objectives: automatic and interactive manual annotation. Indeed, Benato et al. (2021) showed that working on a lower-dimensional space allows to obtain better automatic and manual annotations than working on the original high-dimensional space. That is why a dimensionality reduction technique is used to project the latent feature space extracted by the encoder of the auto-encoder to a smaller 2D space that can be used for visualization and manual annotation purposes. Following the recommendations of Benato et al. (2018) and Benato et al. (2021), we used t-SNE as the dimensionality reduction technique. This choice is well justified, as we are more interested in the local structure of the data than in the global structure, so t-SNE is preferred over methods such as PCA and UMAP. However, t-SNE has three hyper-parameters: the perplexity, the learning rate and the early exaggeration. Therefore, special care needs to be taken when applying t-SNE. To do this, we perform a grid search over the different hyper-parameters of t-SNE, to obtain different 2D projections of the auto-encoder latent feature space. Furthermore, we need to have some criterion to identify 'good' projections for our task (which is label propagation), because visually it can be difficult to distinguish between two 'good' projections, as they can be very similar. The silhouette score (Rousseeuw, 1987) allows this to be done, because it measures the compactness of each class cluster (*i.e.*, cluster of samples belonging to the same class), as well as their distances with respect to the other class clusters. Moreover, for label propagation using KNN strategies, it is ideal to increase the inter-cluster distance and to reduce the intra-cluster distance, because annotation errors mainly come from samples that are located at the boundaries of close class clusters, or from samples located in the wrong class cluster.

Let us now recall the definition of the silhouette score that we propose to use. Let us assume that we have N classes $c_1, \ldots, c_N$, and $f_1, \ldots, f_{L+U}$ embedded representations obtained by a t-SNE projection P. Let us denote for all $j \in [1, L], y_j$ the label of sample j, and for all $i \in [1, N], C_i = \{j \in [1, L] \,/\, y_j = c_i\}$ the set of indices of the samples of class c_i. The silhouette score, S, compares the similarity of a sample $k \in C_p$ between the samples of its own class and the samples of the other classes:

$$S(P) = \frac{1}{L}\sum_{k=1}^{L} s(k)$$

where

$$\forall k \in [1, L], s(k) = \begin{cases} \dfrac{\mu_{inter}(k) - \mu_{intra}(k)}{max(\mu_{inter}(k), \mu_{intra}(k))} & \text{if } |C_p| \geq 2 \\ 0 & \text{else} \end{cases} \quad (1)$$

and where $\mu_{inter}(k)$ is the smallest mean distance between the labeled sample k and all of the labeled samples for the other classes, whereas $\mu_{intra}(k)$ is the mean distance between the labeled sample k and all of the labeled samples of the same class. The best projection P_B is then the one that has the highest silhouette score. It is important to note that only the labeled samples are used to compute the silhouette score as these are the only samples with known labels.

Additionally, as in the previous step, all of the training data (*i.e.*, labeled and unlabeled) are used to optimize the method, because we are interested in data annotation and we separate this task from the classification task. Indeed, the reconstruction part of the auto-encoder model is only used to learn representations (*i.e.*, the training step of the auto-encoder) and the t-SNE algorithm is only used to project the learned representations onto a 2D space.

2.3. Automatic label propagation

Our label propagation method is based on the concept of the local quality (lq) of a projected point onto a lower-dimensional space, as introduced by Lueks et al. (2011). In simple terms, this metric measures how well the neighborhood of a sample is preserved when projected from a high-dimensional space (*i.e.*, the

auto-encoder space) onto a lower-dimensional space (*i.e.*, the 2D space obtained by t-SNE); higher values indicate good preservation of the neighborhood of a point, and lower values indicate modifications in the neighborhood of the point. Moreover, we propose to use the local quality lq of a projected point as the selection metric to obtain the labeled samples to be used for label propagation, and the unlabeled samples to annotate during label propagation. The idea is to use the co-ranking framework (Lee and Verleysen, 2009) to quantify how well the global and local structure of a high-dimensional manifold is preserved when projected onto a lower dimension using a dimensionality reduction technique such as t-SNE. The principle is the following. Let us define the rank of sample A with respect to sample B as the index of A in a sorted list (by increasing distance) of the neighbours of B. The idea is to compute the ranks of all of the samples (with respect to all of the other samples) in the high-dimensional manifold and in the lower-dimensional manifold, and to compare them, to quantify how much they change. The hypothesis is that if the neighborhood of a sample is unchanged when it is projected onto the lower-dimensional manifold, then its ranks will also remain unchanged (so ideally the structure assumption (Chapelle et al., 2009) should be verified in both spaces). The computation of the global and local quality metrics introduced by Lueks et al. (2011) depends on two parameters: k_s and k_t. Here, k_s controls the size of the neighborhood that we are going to look for when comparing the neighborhoods[2] in the high- and low-dimensional manifolds. Then, k_t controls the rank error (*i.e.*, the rank changes that we are going to tolerate[3]).

Furthermore, once we have computed the global and local qualities of the selected projection P_B, we can start using these to propagate labels from labeled samples to unlabeled samples. To formalize our method, we are going to build on (Zhu and Ghahramani, 2002). Let us denote $Y \in R^{(L+U,N)}$ as the label matrix, where the first L rows correspond to the labeled samples and the last U rows correspond to the unlabeled samples. As we work with probabilistic labels, Y_{ij} is the probability that sample i belongs to class j. Let us denote, $T^{K,\tau} \in R^{(L+U,L+U)}$ as a probabilistic transition matrix, where $T_{ij}^{K,\tau}$ is the probability to jump from sample i to sample j, which depends on K, the size of the neighborhood used to search for labeled neighbours for an unlabeled sample (not to be confused with k_t, which was used before to compute the local quality of the 2D points), and τ, the threshold used to determine whether the local quality of a point is considered acceptable or not (not to be mistaken for k_s). We define $T^{K,\tau}$ based on the nearest-neighbors method and the local quality:

$$\forall i, j \in [1, L+U], T_{ij}^{K,\tau} = \begin{cases} 1 & \text{if } (i, j \in [1, L] \text{ and } i = j) \\ & \text{or } (i, j \in \mathcal{P}_{K,\tau}) \\ & \text{or } (i = j \text{ and } i \in \mathcal{C}_\tau) \\ 0 & \text{else} \end{cases} \quad (2)$$

with

- $P_{K,\tau} = \{i \in [L+1, L+U], j \in [1, L] \text{ s.t. } f_i \in \mathcal{V}_K(f_j), lq(f_i, k_s, k_t) > \tau, lq(f_j, k_s, k_t) > \tau, \forall f \in \mathcal{V}_K(f_i), lq(f_j, k_s, k_t) > lq(f, k_s, k_t)\}$, where
  - $f_i \in \mathcal{V}_K(f_j)$ means that the embedded representation of the unlabeled sample i is in the K-neighborhood of the embedded representation of the labeled sample j;
  - $lq(f_i, k_s, k_t) > \tau, lq(f_j, k_s, k_t) > \tau$ means that the local quality of samples i and j are greater than the defined threshold τ;
  - $\forall f \in \mathcal{V}_K(f_i), lq(f_j, k_s, k_t) > lq(f, k_s, k_t)$ means that the embedded representation of the labeled sample j is the one that has the best local quality in the K-neighborhood of the unlabeled sample i.
- $\mathcal{C}_\tau = \{i \in [L+1, L+U] \text{ s.t. } lq(f_i, k_s k, k_t) < \tau\}$ is a set containing all of the unlabeled samples with a local quality score smaller than the defined threshold τ. These samples will not be taken into account for label propagation.

The set $P_{K,\tau}$ allows propagation of the labels from the labeled samples to their unlabeled neighbors based on a local quality criterion, while the set C_τ avoids labeling samples that do not respect the local quality criterion. We can now define our label propagation algorithm as in Zhu and Ghahramani (2002):

- Propagate the labels from the good local quality labeled samples to the good local quality unlabeled samples: $Y \leftarrow T^{K,\tau} \times Y$;
- Row normalize Y (by construction of $T^{K,\tau}$, Y is row-normalized);
- Update $T^{K,\tau}$ by considering adding the new labeled samples to $\mathcal{L}$;
- Repeat the process until there are no more samples to label (or until some number of iterations is reached).

Due to the formalism introduced in Zhu and Ghahramani (2002) and used here, we can see the difference between our introduced method and the method introduced by Benato et al. (2018, 2021): in our method, the transition matrix T is computed through KNN and local quality measures, whereas with Benato et al. the transition matrix T is computed using Laplacian SVM and OPF.

Intuitively, our algorithm finishes when there are no more samples to label, or when there are no more unlabeled samples with local quality greater than the established threshold. The final algorithm of our method is presented in Algorithm 1.

Algorithm 1: Local quality with KNN (LQ-KNN) label propagation.

Input: $\mathcal{D} = \mathcal{L} \cup \mathcal{U}$, k_s, k_t, K, τ
Output: New labeled dataset $\tilde{\mathcal{D}}$
Iterations:

- Extract features of ALL of the samples using an auto-encoder model.
- Dimensionality reduction of the previous representations:
  - Apply t-SNE with grid search;
  - Select the best projection P_B using the silhouette score;
  - Obtain the embedded representations $f_1, \ldots, f_{L+U}$ of the samples using P_B;
  - Compute the local quality $lq(., k_s, k_t)$ of each sample;
  - Sort the representations obtained by decreasing the local quality.
- Propagate the labels using the local quality of the embedded representations:

while $P_{K,\tau} \neq \emptyset$ **do**
  $Y \leftarrow T^{K,\tau} \times Y$;
  Row normalize Y;
  Update $\mathcal{L}$, $\mathcal{U}$, $T^{K,\tau}$ and $P_{K,\tau}$;
end

- Define $\tilde{\mathcal{D}} = \mathcal{L}$

[2] the higher k_s, the more demanding it is in terms of global quality

[3] the higher k_t, the more errors we tolerate, but the less informative in the local quality

2.4. Classification with noisy labels

Once we obtain a new labeled dataset with our proposed method, we perform another task: naming classification, to take advantage of having more labeled data. However, as expected, our method introduces some noise into the labels, which can disrupt the learning process of the classification model. To overcome this, we propose to use one noise robust loss function, GCE loss (Zhang and Sabuncu, 2018), which behaves well under noisy-label situations, and allows the trained models to maintain good generalization performances for unseen data. If we note, for all samples $x \in \mathcal{L}$ of label $y \in \{0,1\}^C$, $g(x)$ the prediction of a classification model, the GCE loss is defined as:

$$\mathcal{L}_q(g(x), y_i) = \frac{1 - g_i(x)^q}{q} \tag{3}$$

where y_i and $g_i(x)$ are the i-th components of the true label y and the predicted label $g(x)$. The hyper-parameter q allows control of the noise tolerance and the convergence speed; when $q \to 1$, we get (ignoring a multiplication factor) the mean absolute error loss function, which is known to be noise tolerant but with slow convergence speed, whereas when $q \to 0$, we get the CE loss function, which is known to have fast convergence speed but which is not noise tolerant. Moreover, following the recommendations of (Zhang and Sabuncu, 2018), we are going to fix $q = 0.7$, as this represents a good trade-off between noise tolerance and convergence speed.

3. Experiments and results

In order to validate our proposed method, we test it on three different datasets: MNIST (LeCun and Cortes, 2010), OrganCMNIST (Yang et al., 2021; Bilic et al., 2019) and a private dataset composed of TCD HITS. Without loss of generality, the models used for feature extraction and classification tasks were adapted to each dataset. The core of our method (dimensionality reduction, label propagation, robust loss function) is applied using the same parameters for all the datasets.

3.1. Data

We used two public datasets, MNIST and OrganCMNIST. The first one is a subset of the MNIST dataset (LeCun and Cortes, 2010), which includes 15,000 labeled samples for training, and 10,000 labeled samples for testing. The second one is the OrganCMNIST (Yang et al., 2021; Bilic et al., 2019) dataset, which includes 15,392 labeled samples for training and 8,268 labeled samples for testing. This dataset is composed of 28×28 computed tomography images of 11 different organs.

The HITS dataset includes 52 patients (20 men, 25 women, and 7 unknown; median age 69, range 21 to 91, computed with the available information) from 11 hospitals from France, Switzerland, Belgium, England, and The Netherlands. Some of the patients were on Neurovascular Units and Cardiovascular Units, and we identified two pathologies: stenosis and patent foramen ovale. Additionally, the data of some of the patients were acquired during surgical procedures: transcatheter aortic valve implantation and atrial fibrillation ablation. Furthermore, some of the patients received a contrast agent during the recording, which were mainly Sonovue (an ultrasound contrast agent) for the hospitals in Lyon (France), and Iobitridol (an iodine-containing contrast agent) for the hospitals in Belgium.

The recordings were acquired using two TCD devices (TCD-X, WAKIe; Atys Medical) under different conditions and for different durations, and the recording conditions and parameters were different across all of the patients. However, according to the device settings ranges and the recommendations for monitoring the middle cerebral artery and performing emboli detection, we have the following information:

- Pulse repetition frequency: 6.2 kHz;
- Transmitted ultrasound frequency: 1.5 MHz;
- Insonation depth: $45 - 55$ mm;
- Sample volume: $8 - 10$ mm^3.

The data obtained from the TCD recordings, as the raw data, were then processed to obtain suitable representations for the models we developed.

3.2. Pre-processing

From each TCD recording, we detect HITS and extract images (using the data management software, ADMS; Atys Medical), which represents the HITS spectrograms. The detection and extraction parameters were fixed (and were equal for all of the samples in the database). We used a high-pass filter of 150 Hz, a detection threshold for the HITS of 9 dB, a gain of 6 dB, and no noise reduction (the question of the value of the detection threshold was reported in Guepie et al. (2019)). In summary, a HITS was detected if it satisfied the criteria of (Spencer et al., 1995) and if its signal intensity was greater than 9 dB. This procedure provided 68 492 HITS in total, from which 1545 were manually labeled by an expert (403 artifacts, 569 gaseous emboli, 569 solid emboli, 4 unknown) using the 2D reduced space obtained using our pipeline (figure 2). This labeled dataset is henceforth referred to as the HITS dataset whereas the partially labeled dataset (68 492 HITS, 1545 labeled) is referred to as the large HITS dataset.

3.3. Baselines

For each label propagation experiment, we used a standard KNN strategy (Std-KNN) (Vindas et al., 2021), where the labels are propagated using only a KNN algorithm without local quality (so the computation of the transition matrix T^K depends only on K, the neighborhood considered to propagate the labels). Additionally, we compare our method (LQ-KNN) to OPF-semi (Amorim et al., 2014), which is commonly used for data annotation. We use the python library OPFython (de Rosa and Papa, 2021) to implement the OPF-semi models. For the classification experiments, our baselines are: (1) models that use only the original labeled dataset; and (2) models that use the augmented dataset trained without noise-tolerant loss functions.

3.4. Evaluation strategy

*Evaluation of label propagation*To evaluate our label propagation method, we used the annotation accuracy, which is defined as the ratio between the number of correctly labeled samples and the total number of labeled samples provided by the method. We also used the percentage of labeled samples, which is defined as the ratio between the number of automatically labeled samples and the initial number of unlabeled samples. For label propagation experiments using a fully annotated dataset, we considered only 10% of the training samples as originally labeled (*i.e.*, $\mathcal{L}$ is composed of 10% of the available labeled samples, and $\mathcal{U}$ is composed of the remaining samples). Then, we select the optimal projection using only the labeled samples and we propagate the labels from these samples to the rest of the samples using the selected projection. The selection of the originally labeled samples is carried out using random sampling. For statistical purposes, we repeat each label propagation run 50 times. In addition to these 50 repetitions, for the MNIST experiment, we train 10 auto-encoder models, to get tighter statistical results.

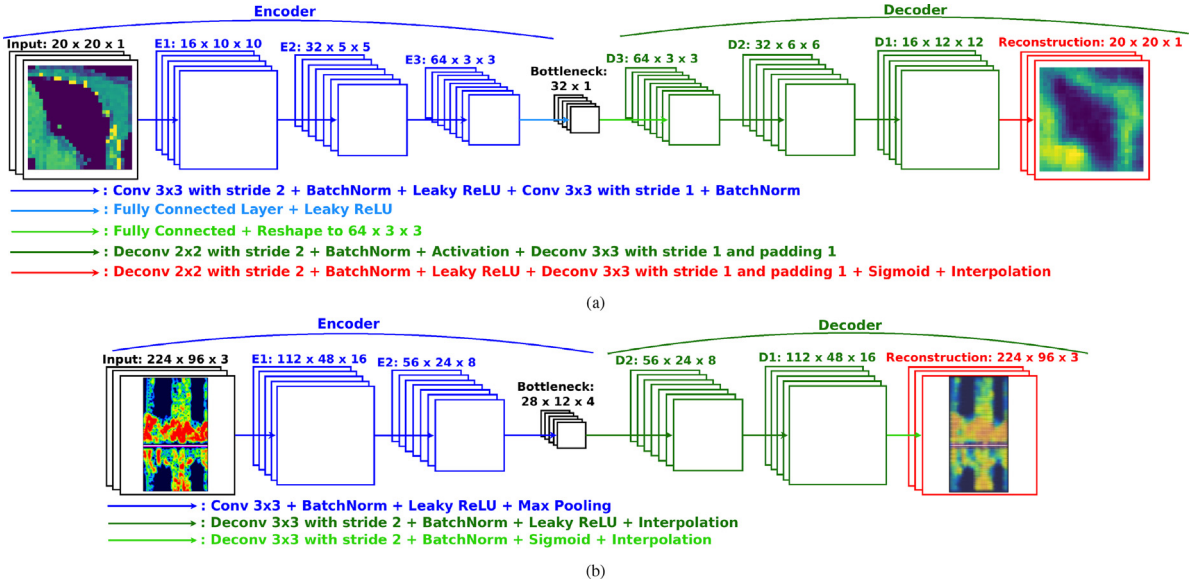


Fig. 3. Auto-encoder architectures used. (a) Architecture for the MNIST and OrganCMNIST datasets. (b) Architecture for the HITS dataset.

Table 1
Training parameters of the auto-encoders of experiment 1. MSE stands for mean squared error loss.

Dataset	Epochs	Batch Size	Learning rate	Optimizer	Weight Decay	Loss function
MNIST	50	32	5e-2	1e-5	Adamax	MSE
OrganCMNIST			5e-5	1e-7		
HITS			5e-3	1e-2		

3.4.1. Evaluation of the classification task

Furthermore, we also evaluate the impact of our label propagation method through the performance change of a classifier trained on datasets obtained without and with label propagation. Moreover, we use robust loss functions to compensate for the noise in the labels introduced by the label propagation methods. Two different evaluation strategies were used based on the dataset type. For the MNIST and OrganCMNIST datasets we used 50 times repeated holdout as evaluation method with general accuracy as evaluation metric. The training set was obtained by propagating the labels from 10% of the training samples (15,000 for MNIST and 15,392 for OrganCMNIST) to the rest of the training samples, whereas the test set was fixed and composed of the manually labeled testing samples (10,000 for MNIST and 8,268 for OrganCMNIST) which are not used for label propagation. For the HITS datasets, we used the Matthews correlation coefficient (MCC) (Hicks et al., 2021) and class accuracies as evaluation metrics[4] and the evaluation strategy was based on leave-one-subject-out evaluation. First, we propagate the labels from the labeled samples to the unlabeled samples (for the HITS dataset we considered 10% of the samples as labeled and for the large HITS dataset we used all the labeled samples i.e. 1545 samples). Secondly, different train/test splits are created by taking as test samples the manually labeled samples of a fixed subject and as train samples all the (manually and automatically) labeled samples of the remaining subjects. In this way, we get 39 train/test splits. Finally, we train and evaluate different models using the created splits and we repeat this process 10 times for the large HITS dataset and 20 times for the HITS dataset.

[4] We use class accuracy as the noise that we introduce in the labels is asymmetric, mainly between the gaseous emboli and solid emboli classes

Moreover, for all of the experiments using local quality metrics, we fixed $k_s = 10$ and $k_t = 10$. We choose these values because: (1) we saw experimentally that the values of k_s and k_t do not have an important influence on the annotation accuracy for values that are not too large (*i.e.*, less than 50); (2) we want good local qualities in small neighborhoods to better propagate labels; (3) higher values of k_s and k_t could lead to 'false' good local-quality points. Further discussion can be found in Section 4.

3.5. Experimental set-up

3.5.1. Experiment 1: Label propagation evaluation

The objective of this experiment was to test our method using three datasets: a subset of the MNIST dataset verifying the structure assumption; and two medical datasets, as the OrganCMNIST and the HITS datasets. The auto-encoder architectures used for unsupervised feature extraction are shown in Fig. 3 and its training parameters are given in Table 1. As regards the optimal projection selection, we did the grid search over three hyper-parameters (perplexity, early exaggeration and learning rate) and their ranges can be found in Table 2. Finally, the parameters of the different label propagation experiments are given in Table 3, and the results are given in Table 4 and shown in Figs. 4, 5, and 14. Several phenomena can be observed.

First, from Table 4, we can see that our method, LQ-KNN, is comparable to OPF-semi. Indeed, although Std-KNN and LQ-KNN do not annotate all the available samples for the hyper-parameters tested (contrary to OPF-semi), they annotate more than 96% of the unlabeled samples with an annotation accuracy greater than 90% for the MNIST dataset, 79% for the OrganCMNIST dataset, and 81% for the HITS dataset compared to 82%, 75% and 78% for OPF-semi for the MNIST, OrganCMNIST and HITS datasets, respectively. Additionally, we can see that our method is faster than OPF-semi, by a factor of $10^2 - 10^3$.

Table 2
Parameters for the grid search in experiment 1. As the HITS and OrganCMNIST datasets are more complex than the MNIST dataset, a more complete grid search is needed to find optimal projections.

Dataset	Perplexity	Early Exaggeration	Learning rate
MNIST	[10, 30, 50]	[50, 250, 500]	[10, 100, 1000]
OrganCMNIST	[5, 10, 15, 20, 25, 30, 35, 40, 45, 50]	[5, 10, 25, 50, 75, 100, 200, 500]	[10, 50, 100, 500, 1000]
HITS			

Table 3
Parameters for label propagation in experiment 1. The Dataset corresponds to the dataset used as the basis to test the label propagation method. We select 10% of the samples as labeled ($\mathcal{L}$), and we consider the rest of the samples as unlabeled ($\mathcal{U}$). We then propagate the labels from the samples of $\mathcal{L}$ to some of the samples of $\mathcal{U}$ using one of the propagation methods, to obtain the final dataset. The experiments using the HITS and OrganCMNIST datasets were repeated 50 times. The experiments using the MNIST dataset were repeated 50 times for 10 different auto-encoders (500 repetitions in total), except for OPF-semi, where we did 20 repetitions for each auto-encoder. K corresponds to the size of the neighborhood used to search for labeled neighbours for an unlabeled sample.

Exp. name	Dataset	$\lvert\mathcal{L}\rvert$	$\lvert\mathcal{U}\rvert$	Propagation	K	τ	Repetitions
Std-KNN	MNIST	1,496	13,504	Std-KNN	$1 \leq K \leq 20$	-	500
	HITS	152	1,393			-	50
	OrganCMNIST	1,534	13,858			-	50
LQ-KNN-τ	MNIST	1,496	13,504	LQ-KNN		$0.1 \leq \tau \leq 0.5$	500
	HITS	152	1,393				50
	OrganCMNIST	1,534	13,858				50
OPF-semi	MNIST	1,496	13,504	OPF-semi	-	-	200
	HITS	152	1,393		-	-	50
	OrganCMNIST	1,534	13,858		-	-	20

Table 4
Experiment 1: Label propagation results using the MNIST, OrganCMNIST and HITS datasets. $\mathcal{L}$ corresponds to the set of initially (manually) labeled samples, $\mathcal{U}$ corresponds to the set of initially unlabeled samples, τ corresponds to the local quality threshold that defines if a sample is considered as of good quality, K corresponds to the size of the neighborhood used to search for labeled neighbours for an unlabeled sample. Our proposed method LQ-KNN outperforms OPF-semi (Amorim et al., 2014) and the baseline Std-KNN, at the expense of a smaller number of labeled samples. Additionally, LQ-KNN and Std-KNN are faster than OPF-semi by a factor of 10^3.

Dataset	Propagation method	$\lvert\mathcal{L}\rvert$	$\lvert\mathcal{U}\rvert$	τ	K	Annotation accuracy	Final % of labeled samples (%)	Annotation time (ms/sample)
MNIST	Std-KNN	1496	13,504	-	5	91.83 ± 1.47	95.39 ± 1.05	$(30.98 \pm 5.84) \times 10^{-3}$
				-	10	90.74 ± 1.45	99.43 ± 0.23	$(\mathbf{28.78 \pm 5.13}) \times \mathbf{10^{-3}}$
	LQ-KNN			0.1	5	$\mathbf{93.12 \pm 1.36}$	93.88 ± 0.66	$(59.10 \pm 12.35) \times 10^{-3}$
					10	92.66 ± 1.30	98.16 ± 0.42	$(50.48 \pm 11.32) \times 10^{-3}$
	OPF-semi			-	-	82.32 ± 6.17	$\mathbf{100.0 \pm 0.0}$	102.71 ± 17.52
OrganCMNIST	Std-KNN	1534	13,858	-	5	81.87 ± 0.76	90.26 ± 2.64	$(26.33 \pm 2.65) \times 10^{-3}$
				-	10	79.86 ± 0.67	99.00 ± 0.20	$(\mathbf{23.41 \pm 1.98}) \times \mathbf{10^{-3}}$
	LQ-KNN			0.1	5	$\mathbf{84.46 \pm 0.57}$	85.62 ± 1.99	$(53.00 \pm 7.47) \times 10^{-3}$
					10	82.73 ± 0.44	96.24 ± 1.09	$(44.36 \pm 5.69) \times 10^{-3}$
	OPF-semi			-	-	75.22 ± 4.48	$\mathbf{100.0 \pm 0.0}$	86.52 ± 0.51
HITS	Std-KNN	152	1393	-	5	82.12 ± 2.37	95.99 ± 1.70	$(10.39 \pm 0.20) \times 10^{-2}$
				-	10	81.36 ± 1.81	99.58 ± 0.63	$(\mathbf{10.04 \pm 0.18}) \times \mathbf{10^{-2}}$
	LQ-KNN			0.1	5	$\mathbf{82.84 \pm 2.12}$	94.48 ± 1.72	$(16.87 \pm 0.48) \times 10^{-3}$
					10	82.67 ± 2.02	98.50 ± 0.80	$(16.13 \pm 0.35) \times 10^{-2}$
	OPF-semi			-	-	78.40 ± 13.44	$\mathbf{100.0 \pm 0.0}$	9.48 ± 1.1

Secondly, from Table 4, we can see that our proposed method (LQ-KNN) yields an annotation accuracy greater than 92% for MNIST, 82% for OrganCMNIST and 82% for the HITS dataset, which is 10%, 7%, and 4% greater than with OPF-semi respectively. Furthermore, the methods using the local quality with neighborhood propagation yield better results than using just Std-KNN. Indeed, with the HITS dataset, Std-KNN with $K = 10$ gives an annotation accuracy of 81.36%, with labeling of 99.58% of the samples, against an annotation accuracy of 82.67% and labeling of 98.50% of the samples for LQ-KNN with $\tau = 0.1$ and $K = 10$. This can also be observed for the MNIST and OrganCMNIST datasets.

Thirdly, from Figs. 4 and 14, we can see that the annotation accuracy and the proportion of labeled samples depends on the neighborhood K within which we propagate the labels; the higher K, the more samples we annotate, but the smaller the annotation accuracy; inversely, the smaller K, the fewer samples we annotate, but the greater the annotation accuracy. Moreover, these quantities depend on the local quality threshold τ used to define good quality samples: the higher τ, the higher the annotation accuracy, but the smaller the number of labeled samples.

Fourthly, in Fig. 4, we can identify two regimes in the behavior of our method. The first regime, which we term the 'dynamic' regime, is obtained at the beginning for relatively small values of K, where the number of newly labeled samples increases with the value of K. The second regime, which we term the 'permanent' regime, is obtained for higher values of K, and in this case the number of labeled samples and the annotation accuracy reach plateaus.

Finally, we studied the importance of the label propagation order by fixing a projection and propagating the labels using LQ-KNN with and without sorting the samples by decreasing local qualities. Fig. 5 shows that the LQ-KNN method that starts by labeling the samples with higher local qualities yields better annotation accuracies than the methods that do not take into account the an-

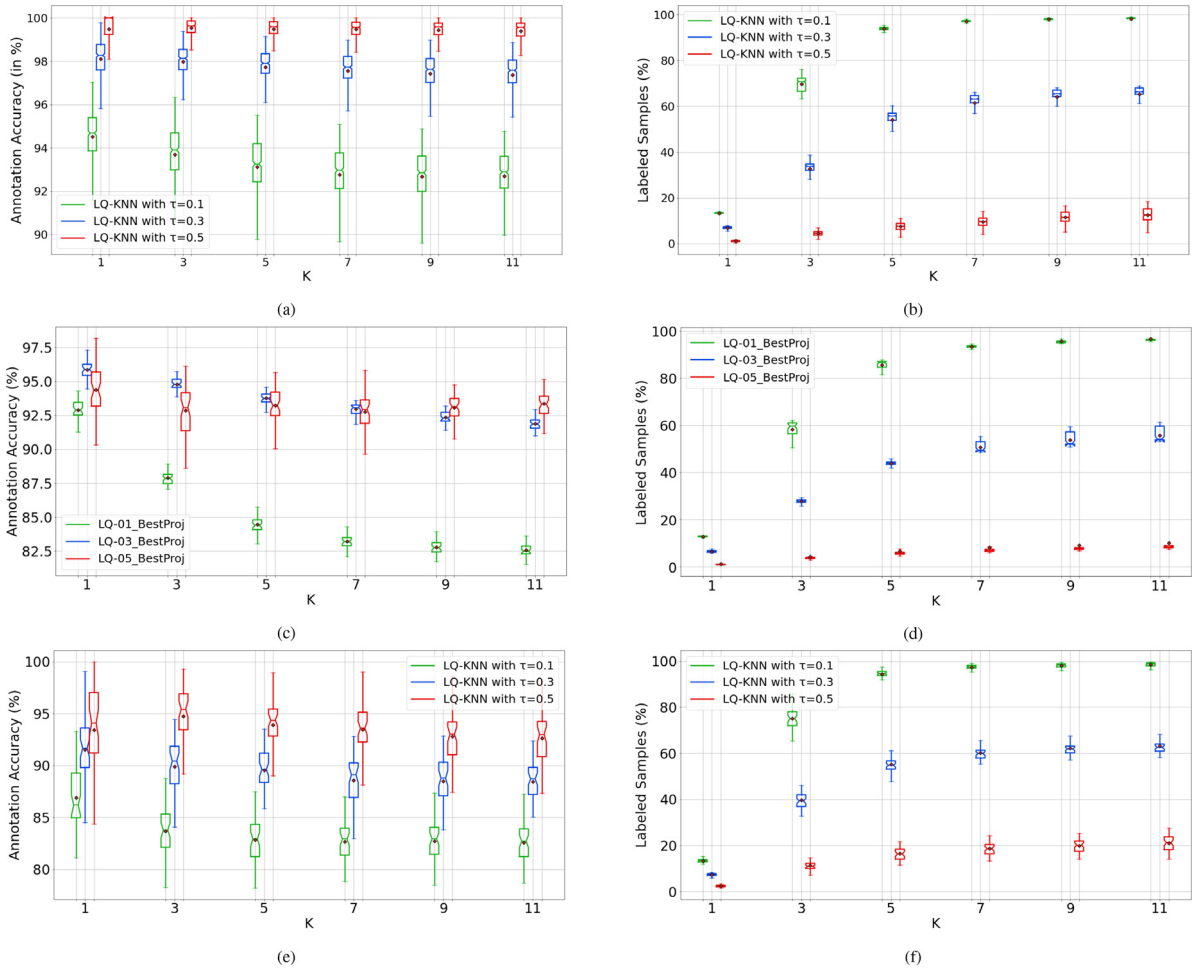


Fig. 4. Experiment 1: Comparing LQ-KNN propagation with different hyper-parameters. (a) MNIST dataset annotation accuracy. (b) MNIST dataset labeled samples (in %). (c) OrganCMNIST dataset annotation accuracy (in %). (d) OrganCMNIST dataset labeled samples (in %). (e) HITS dataset annotation accuracy. (f) HITS dataset labeled samples (in %). τ corresponds to the threshold used to define good local-quality samples. For LQ-KNN: green curves, $\tau = 0.1$; blue curves, $\tau = 0.3$; red curves, $\tau = 0.5$. The proportion (%) of unlabeled samples that were labeled by the methods converges with K, hence we show here the results for $K \leq 11$.

notation order. The difference between the two methods becomes more pronounced when we increase the neighborhood size K used for label propagation.

3.5.2. Experiment 2: Validation of the projection selection strategy

The objective of this experiment is to validate our proposed projection selection strategy (see Section 2.2). To do this, we start by selecting the bests and worsts 2D projections obtained with t-SNE according to the silhouette scores (Fig. 6). Here, the selected best 2D projections have a silhouette score of 0.54, with the worsts at −0.23. Then, we propagate the labels using the same strategy as in experiment 1 for the HITS dataset. These results are given in Table 5. We can see that both propagation methods achieve considerably higher annotation accuracies for the best projection compared to the worst projection for all values of K. Furthermore, even for the worst projection, LQ-KNN provides higher annotation accuracies at the expense of the number of labeled samples.

3.5.3. Experiment 3: Evaluation through a classification task on a dataset with known label noise

We evaluate the results of the previous experiment on a classification task. The objectives of this experiment are two-fold: to determine the improvement in the classification performances through to the use of new automatically labeled data; and to show the interest in using robust loss functions to compensate for the annotation error from the automatic label propagation. We trained a CNN (Fig. 7) on different datasets (see Table 6) with different training parameters based on the dataset (see Table 7). Fig. 8 shows the MNIST dataset results, Fig. 9 shows the OrganCMNIST dataset results, and Figs. 10 and 11 show the results using the HITS datasets.

On the one hand, from the OrganCMNIST results, three interesting points can be noted. First, from Fig. 9, we can see that the best classification performances are achieved with the dataset obtained with our label propagation method, LQ-KNN, and trained with a robust loss function, GCE, which yields a global accuracy of 75.76% against 74.58% with OrganCMNIST Std-KNN and GCE, and 70.62% without label propagation and GCE. Secondly, Fig. 9 also shows that the best results are obtained with the GCE loss function with label propagation (Std-KNN or LQ-KNN). Finally, we can see that even when we do not use a robust loss function (*i.e.*, when we use CE), our label propagation method provides better performances (72.73%) than the baseline OrganCMNIST Std-KNN (71.71%). The same behaviour is observed in Fig. 8 for the MNIST dataset.

On the other hand, the classification results for the HITS dataset (Figs. 10, 11) also reveal the following. First, LQ-KNN and Std-KNN label propagation improve the performances of the model with respect to the model trained using less labeled samples, for all of the classes (in terms of MCC, HITS LQ-KNN-K10 with CE outperforms

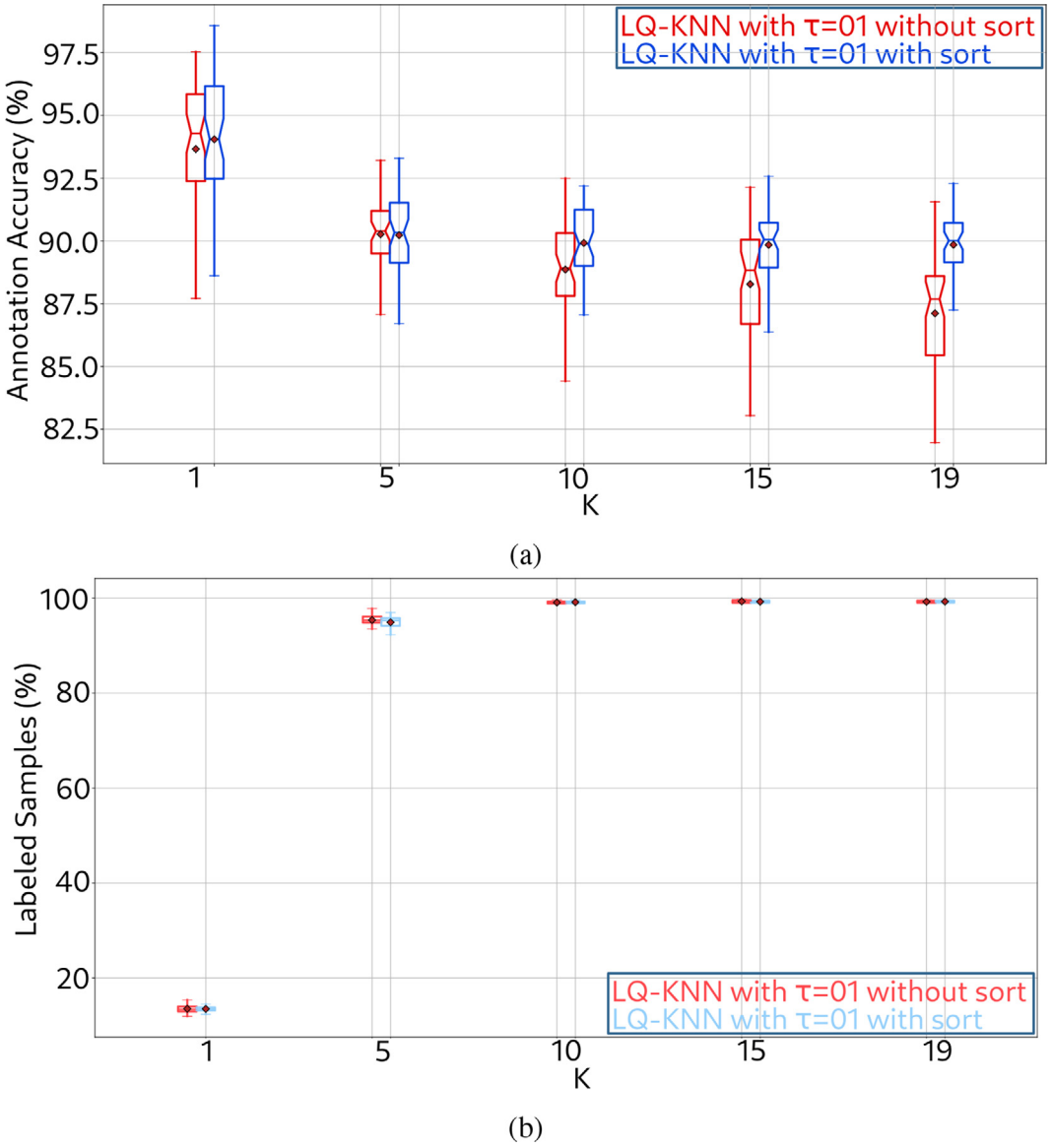
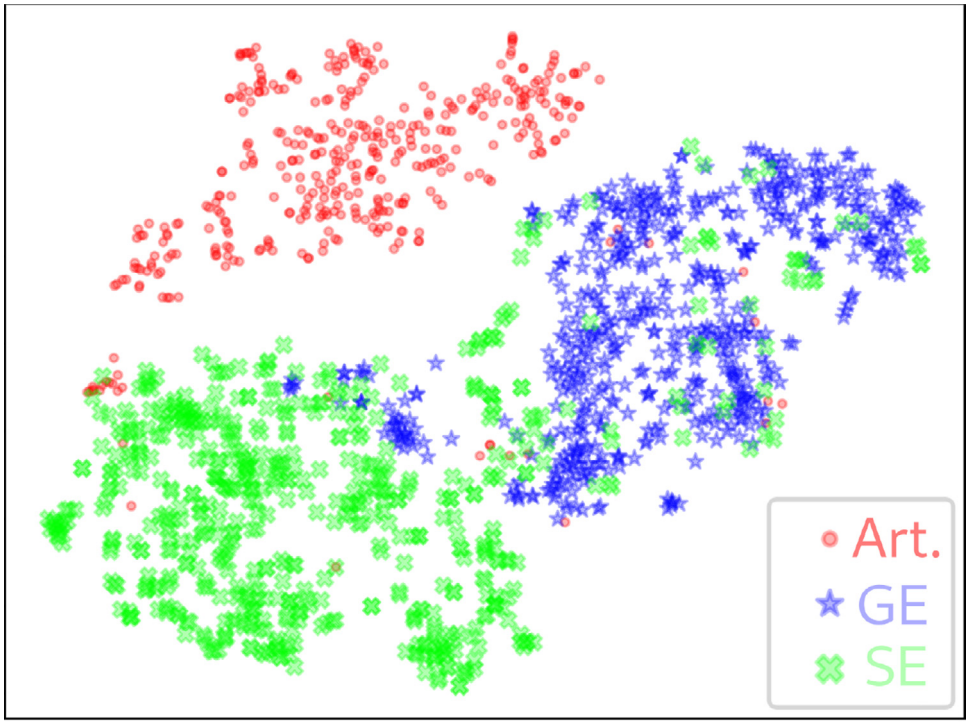


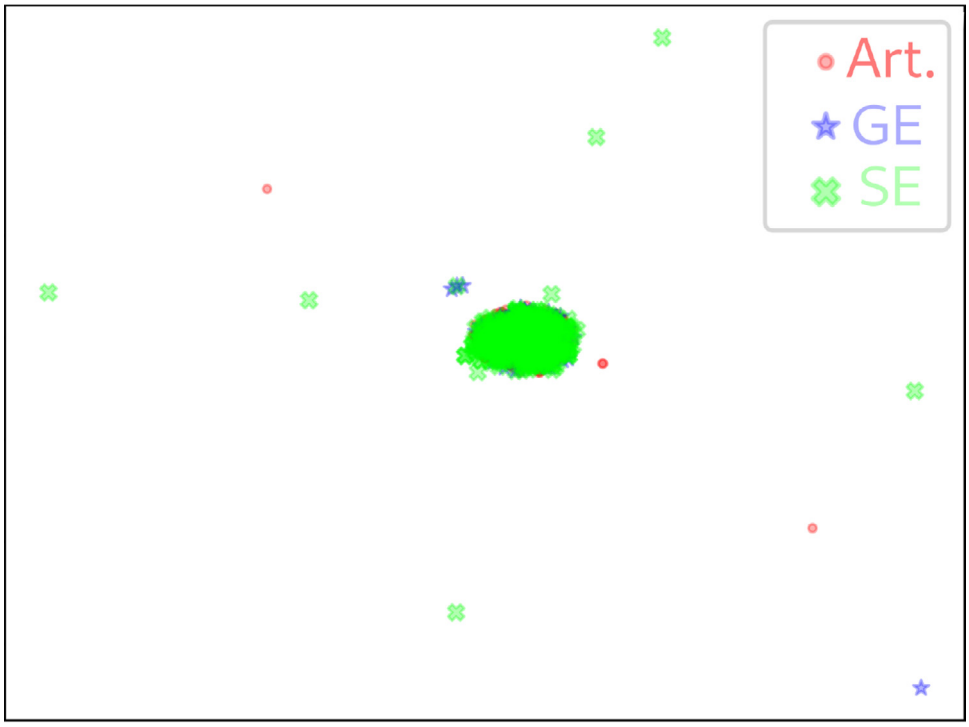
Fig. 5. Experiment 1: Evaluation of the propagation order. (a) Annotation accuracy (in %). (b) Labeled samples (in %). For LQ-KNN with $\tau = 0.1$: blue curves, starting by labeling the higher local quality samples (the samples are sorted by decreasing local qualities); red curves, without taking into account the propagation order (the samples are not sorted by decreasing local qualities). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 5
Experiment 2. Label propagation for the HITS dataset using the best and worst selected 2D projections according to the silhouette scores. The best selected projection allows automatic annotation of more samples with higher annotation accuracy. $\mathcal{L}$, set of initially (manually) labeled samples; $\mathcal{U}$, set of initially unlabeled samples; τ, local quality threshold that defines if a sample is considered as of good quality.

K	Propagation method	Projection	Silhouette Score	$\lvert\mathcal{L}\rvert$	$\lvert\mathcal{U}\rvert$	τ	Annotation accuracy	Final labeled samples %
5	**Std-KNN**	**Best**	$\mathbf{0.53 \pm 0.05}$			**-**	$\mathbf{82.11 \pm 2.37}$	$\mathbf{95.99 \pm 1.70}$
		Worst	-0.26 ± 0.07				58.43 ± 8.95	98.03 ± 1.22
	LQ-KNN	**Best**	$\mathbf{0.53 \pm 0.04}$	**152**	**1,393**	**0.1**	$\mathbf{82.84 \pm 2.12}$	$\mathbf{94.47 \pm 1.72}$
		Worst	-0.25 ± 0.09				70.87 ± 7.18	68.57 ± 10.16
15	**Std-KNN**	**Best**	$\mathbf{0.53 \pm 0.05}$			**-**	$\mathbf{80.31 \pm 2.03}$	$\mathbf{99.97 \pm 0.07}$
		Worst	-0.26 ± 0.07				56.44 ± 9.83	99.66 ± 0.20
	LQ-KNN	**Best**	$\mathbf{0.53 \pm 0.04}$			**0.1**	$\mathbf{82.82 \pm 1.96}$	$\mathbf{98.84 \pm 0.67}$
		Worst	-0.25 ± 0.09				66.16 ± 8.76	79.92 ± 7.531



(a)



(b)

Fig. 6. Experiment 2: Examples of best and worst 2D chosen projections of the HITS dataset (1545 samples) obtained with respect to the silhouette scores. (a) Best projection (silhouette score, 0.54 ± 0.05). (b) Worst projection (silhouette score, -0.23 ± 0.09). The best selected projection gives more distinct clusters per class than the worst. Art., artifact; GE, gaseous emboli; SE, solid emboli.

Table 6
The different datasets used to train the models in experiment 3. The Core dataset corresponds to the dataset used as the basis to test the label propagation method. We select 10% of the samples as labeled ($\mathcal{L}$), and we consider the rest of the samples as unlabeled ($\mathcal{U}$). We then propagate the labels from the samples of $\mathcal{L}$ to some of the samples of $\mathcal{U}$, to obtain the final dataset. K corresponds to the neighborhood that we consider to propagate the labels, and τ corresponds to the local quality threshold that defines if a sample is considered of good quality. None, no labeled propagation used to obtain the final dataset. The mean annotation accuracy correspond to the accuracy of the label propagation method, not to be confused with the classification accuracy obtained by training classification models on these datasets (Figs. 8- 11).

Dataset	Core dataset	Propagation method	$\lvert\mathcal{L}\rvert$	$\lvert\mathcal{U}\rvert$	# of automatically labeled samples	Mean annotation accuracy	**K**	τ
MNIST No propagation	MNIST	None	1496	13,504	-	-	-	-
MNIST Std-KNN		Std-KNN			13426 ± 31	90.74 ± 1.45	10	-
MNIST LQ-KNN		LQ-KNN			13256 ± 56	92.66 ± 1.30	10	0.1
OrganCMNIST No propagation	OrganCMNIST	None	1534	13,858	-	-	-	-
OrganCMNIST Std-KNN		Std-KNN			13720 ± 28	81.87 ± 0.76	10	-
OrganCMNIST LQ-KNN		LQ-KNN			13336 ± 151	82.73 ± 0.44	10	0.1
HITS Whole	HITS	None	1,545	0	-	-	-	-
HITS No propagation			152	1393	-	-	-	-
HITS Std-KNN-K2		Std-KNN			591 ± 42	84.95 ± 2.61	2	-
HITS Std-KNN-K10					1387 ± 8.7	81.36 ± 1.81	10	-
HITS LQ-KNN-K3		LQ-KNN			554 ± 63	89.88 ± 2.77	3	0.3
HITS LQ-KNN-K4					700 ± 54	89.65 ± 2.36	4	0.3
HITS LQ-KNN-K10					1372 ± 11	82.67 ± 2.02	10	0.1

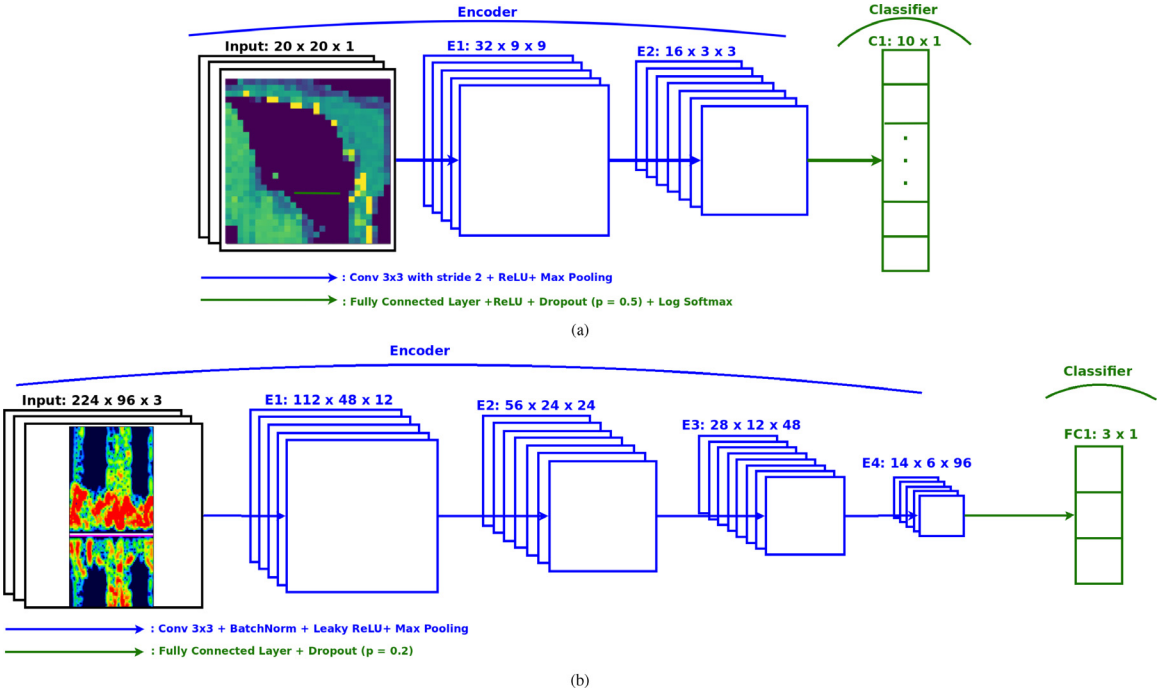


Fig. 7. Convolutional neural network architectures used for classification for the different datasets. (a) Architecture for the MNIST and OrganCMNIST datasets. (b) Architecture for the HITS dataset.

HITS with no propagation with CE by 5.68%, and HITS LQ-KNN-K10 with GCE outperforms HITS with no propagation with GCE by 9.71%).

Secondly, when we propagate the labels to annotate less than 50% of the unlabeled samples (*i.e.*, when we use HITS Std-KNN-K2, and HITS LQ-KNN-K4 datasets in Table 6), LQ-KNN propagation outperforms Std-KNN for both loss functions. Interestingly, we can see that even though HITS LQ-KNN-K3 has fewer labeled samples than HITS Std-KNN-K2, it provides better classification performances, when using a nonrobust loss function. Indeed, the classifier trained on HITS LQ-KNN-K3 with CE outperforms that trained on HITS Std-KNN-K2 with CE by a margin of 1.90% in terms of MCC. This is not observed when using a robust loss function.

Finally, HITS LQ-KNN-k10 CE and HITS LQ-KNN-k10 GCE outperform HITS Std-KNN-k10 GCE and HITS Std-KNN-k10 CE in terms of MCC and class accuracy. For both loss functions, HITS LQ-KNN-k10 outperforms HITS Std-KNN-k10 in terms of solid emboli accuracy by a margin of over 3.5%. This is particularly interesting as solid emboli are the most critical class, because solid emboli can cause ischemic stroke. Additionally, HITS LQ-KNN-k10 GCE performs similarly to HITS Whole CE and HITS Whole GCE, which are fully manually labeled datasets.

3.5.4. Experiment 4: Classification using a semi-automatically labeled HITS dataset with unknown label noise

The objective of this experiment was to study the behavior of our method, LQ-KNN, on a larger-scale real medical dataset with unknown label noise. We used our semi-automatic labeling method to create different datasets using all of the available labeled samples (*i.e.*, the 1,545 samples) and part of the avail-

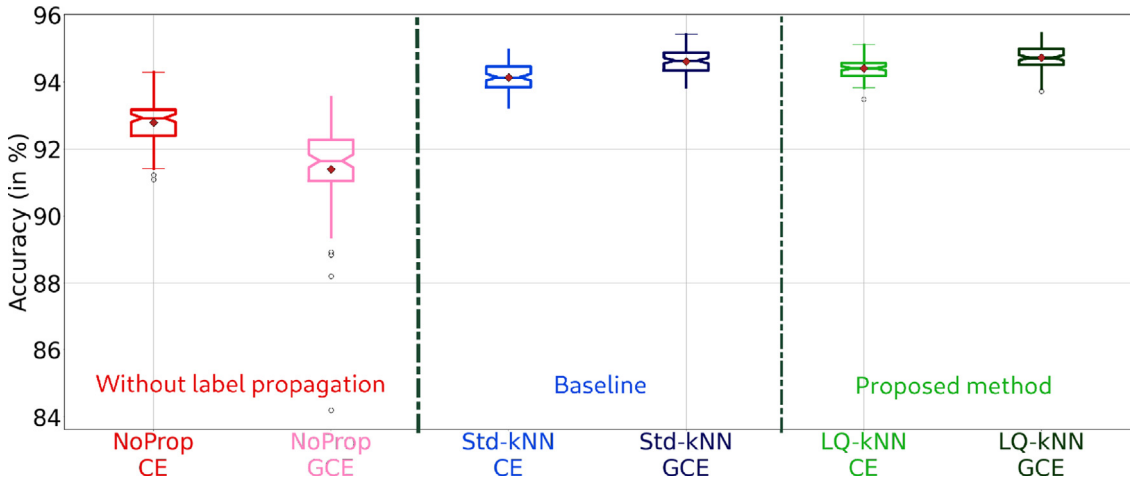


Fig. 8. Experiment 3: Comparison of the accuracy of the different label propagation methods on the MNIST dataset. The best performing classification model is the one trained on the dataset obtained with our proposed method, LQ-KNN, and with a robust loss function. When using nonrobust loss functions, the best performing classifier is the one trained with the dataset obtained using LQ-KNN.

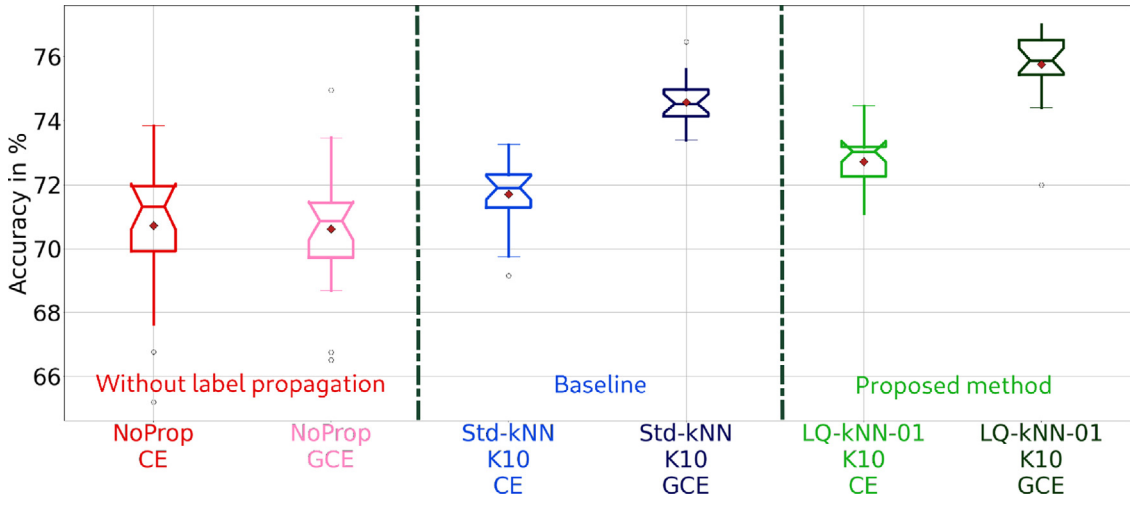


Fig. 9. Experiment 3: Comparison of the test accuracy of the different label propagation methods on the OrganCMNIST dataset. The best performing classification model is the one trained on the dataset obtained with our proposed method, LQ-KNN, and with a robust loss function. When using nonrobust loss functions, the best performing classifier is the one trained with the dataset obtained using LQ-KNN.

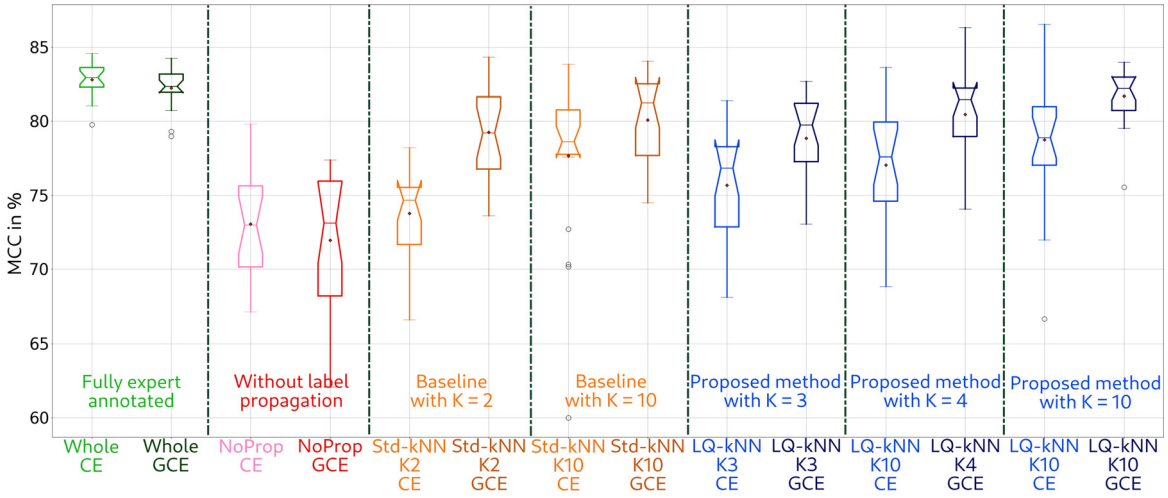


Fig. 10. Experiment 3: Test Matthews correlation coefficient (MCC) as comparisons using the semi-automatically labeled HITS dataset with known label noise. Both label propagation methods increase the classification performances of the trained model, with similar performances to a model trained with a fully manually labeled dataset. Our proposed method, LQ-KNN, globally outperforms the baseline model.

Table 7
The different training parameters used in experiment 3. q represents the hyper-parameter of the GCE loss function giving a trade-off between convergence speed and robustness to label-noise. The higher the value of q, the more robust GCE but the smaller the convergence speed; the smaller the value of q the faster the convergence but the smaller the robustness to label-noise.

Dataset	Epochs	Batch Size	Learning rate	Weight Decay	Optimizer	Loss function	q
MNIST	100	32	7e-3	1e-7		CE	-
						GCE	0.7
OrganCMNIST	150		7e-3	1e-5	Adamax	CE	-
						GCE	0.5
HITS	50		2e-2	1e-7		CE	-
			1e-3			GCE	0.7

Table 8
New datasets used in experiment 4. The objective of the experiment was to study the behavior of our method on a larger-scale real medical dataset with unknown label noise.

Dataset	Propagation Method	$\lvert\mathcal{L}\rvert$	$\lvert\mathcal{U}\rvert$	# of automatically labeled samples	Samples per class	**K**	τ
HITS Std-KNN Large	Std-KNN	1,545	66,947	13,653	4,551	10	-
HITS LQ-KNN Large	LQ-KNN	1,545	66,947	14,970	4,990	10	0.1

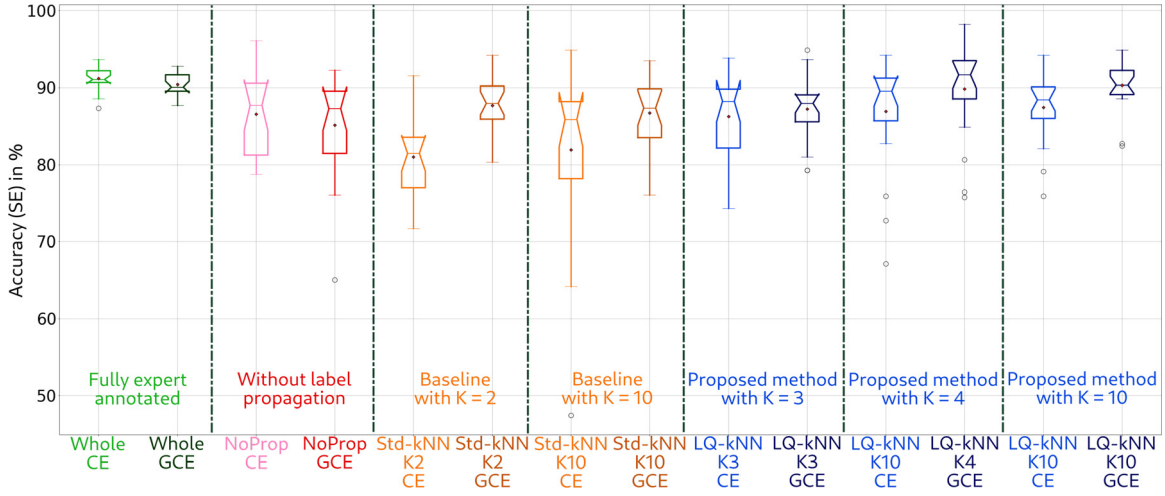


Fig. 11. Experiment 3: Solid emboli test accuracies as a comparison of semi-automatically labeled HITS datasets with known label noise. Our proposed method, LQ-KNN, outperforms the baseline model with the proposed hyper-parameters.

able unlabeled samples (with unknown labels). To avoid imbalanced dataset problems, we balanced the classes based on the final numbers of labeled solid emboli: we sample in the artifacts and gaseous emboli classes to get the same number of samples as the number of labeled solid emboli (this is possible, as we have more artifacts and gaseous emboli than solid emboli). We evaluate the different datasets on a classification task, proceeding in the same way as in experiment 3. To do this, we trained a classification CNN with the architecture in Fig. 7b on two new datasets, HITS Std-KNN Large and HITS LQ-KNN Large, plus the datasets of experiment 3 (see Table 8). The training was carried out using two loss functions, CE and GCE, with a learning rate of $1e-3$, a batch size of 32, a weight decay of $1e-7$, during 50 epochs. The results are shown in Figs. 12 and 13. Three main points can be noted.

First, we can see that HITS LQ-KNN-Large trained with CE and GCE outperforms all of the Std-KNN and LQ-KNN methods in terms of MCC and SE accuracy (including those of experiment 3). If we look at the MCC, HITS LQ-KNN-Large outperforms HITS Std-KNN-Large by a margin greater than 2% (where HITS LQ-KNN-Large GCE outperforms the other datasets). We observe similar behavior for the SE accuracies.

Secondly, we can see that with respect to the results in experiment 3, the performances of the model trained using HITS Std-KNN-Large increases significantly for the solid emboli class, decreases for the artifact and gaseous emboli classes, while the variability is reduced when using more samples. On the other hand, the performances of the model trained using LQ-KNN improve significantly for the solid emboli and gaseous emboli classes, and decrease for the artifact class, while the variability is reduced.

Finally, in terms of MCC, the best performing model for this experiment is obtained using the GCE loss function and the HITS LQ-KNN-Large dataset, which outperforms the best performing model of experiment 3 trained on the HITS Whole dataset that was fully manually labeled and without label noise. However, when using a nonrobust loss function (*i.e.*, CE), HITS LQ-KNN-Large and HITS Whole have similar behaviors, even though HITS LQ-KNN is a larger dataset. Moreover, HITS LQ-KNN-Large improves the general solid emboli accuracy with respect to HITS Whole for both loss functions. This comes at the expense of significant decrease in the artifacts accuracy. Nonetheless, when we use robust loss functions to compensate for the noise (mainly between the gaseous emboli and solid emboli classes) introduced with LQ-KNN, this also outperforms the HITS Whole dataset for the gaseous emboli class.

Implementation details

All of the codes were implemented using Pytorch (Paszke et al., 2019) and Scikit-Learn Pedregosa et al. (2011). The different experiments were carried out on a high-performance computing cluster with 25 heterogeneous machines (each machine with between

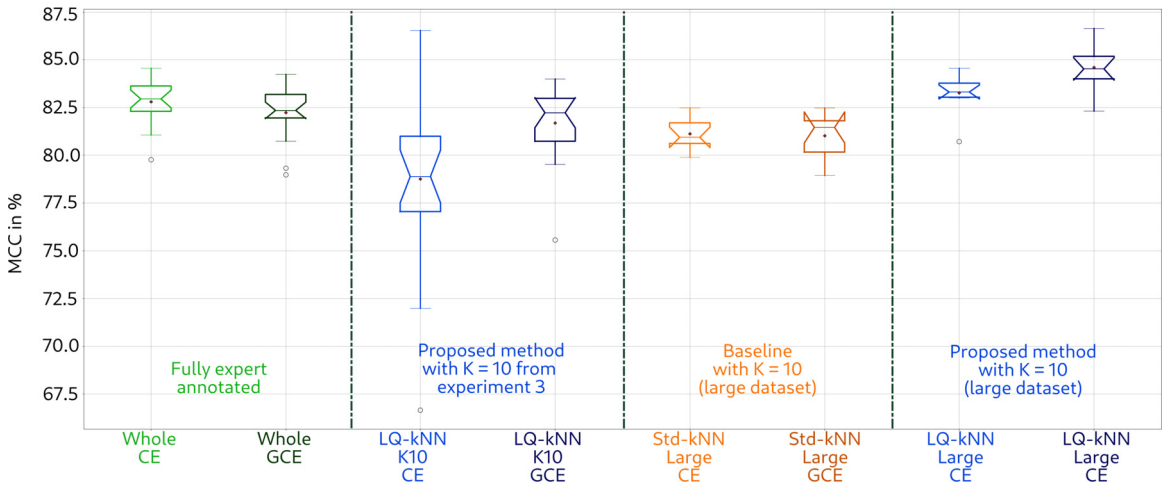


Fig. 12. Experiment 4: Tests Matthews correlation coefficient (MCC) as comparisons using semi-automatically labeled HITS datasets with unknown label noise. We propagate the labels from 1545 manually labeled samples to part of the remaining 66 947 unlabeled samples. Based on the label propagation method used, we obtain datasets with different numbers of samples (however, the classes are always balanced): 15 198 samples when using the baseline Std-KNN with $K = 10$, and 16 515 when using LQ-KNN with $K = 10$ and $\tau = 0.1$.

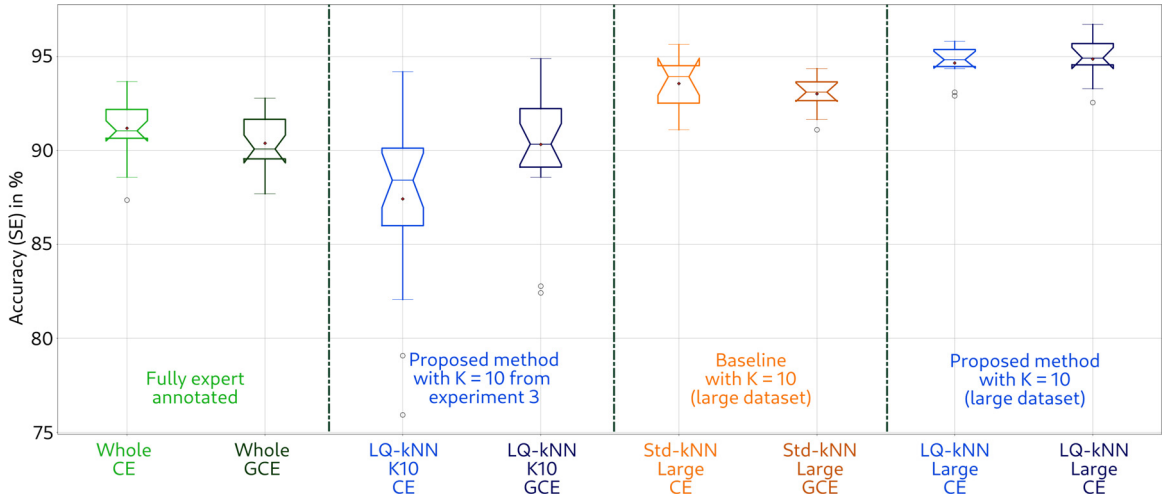


Fig. 13. Experiment 4: Solid emboli test accuracies as comparisons using semi-automatically labeled HITS datasets with unknown label noise. We propagate the labels from 1545 manually labeled samples to part of the remaining 66 947 unlabeled samples. Based on the label propagation method used, we obtain datasets with different numbers of samples (however, the classes are always balanced): 15 198 samples when using the baseline Std-KNN with $K = 10$, and 16 515 when using LQ-KNN with $K = 10$ and $\tau = 0.1$.

16 Gb and 128 Gb of RAM, CPUs with 8 to 32 cores, and different types of Nvidia Quadro RTX and Tesla GPUs). The GitHub for the MNIST experiments can be found at: https://github.com/yamilvindas/LQ-KNN_DataAnnotation.

4. Discussion

Experiment 1

This experiment confirms that our method, LQ-KNN, is comparable to the state-of-the-art, as it outperforms OPF-semi, which is commonly used for data annotation. Indeed, LQ-KNN achieves higher annotation accuracy at the expense of a smaller number of annotated samples (from 1.5% to 3.8% fewer samples labeled by LQ-KNN) for all the tested datasets. Another advantage of our method over OPF-semi is its annotation speed; it is faster than OPF-semi by a factor of $10^2 - 10^3$, which can be nonnegligible when annotating large datasets. This difference in annotation time is explained on the basis that using OPF-semi implies training a classifier using all of the available samples (*i.e.*, labeled and unlabeled), and then predicting the labels of the unlabeled samples using this classifier, which are time-costly tasks. Furthermore, this experiment shows that our method, LQ-KNN, with $\tau = 0.1$ behaves better than the baseline Std-KNN. This is even more evident when we increase τ, to reach annotation accuracies of 99.34% to label 15.51% of samples for $\tau = 0.5$ and $K = 20$ for the MNIST dataset. What is more, we observe smaller annotation accuracies for the HITS dataset for all methods. For Std-KNN and LQ-KNN this can be explained on the basis that gaseous emboli and solid emboli can be easily confused in some cases, such that the two clusters/ manifolds can overlap. Moreover, when we propagate labels using samples that are at the boundary between solid emboli and gaseous emboli, we make more annotation errors. However, as the annotation accuracies show, the use of the local quality of the projection allows the label propagation to be more cautious, so samples that were wrongly projected at the boundary are not used for label propagation. By the same token, this experiment also reveals one key advantage of our proposed method: the annotation error control. Indeed, due to the hyper-parameters of our method, we can reduce the annotation error at the expense of the number of labeled samples. Fig. 14 gives a clear example where by reducing the local quality threshold τ and the neighborhood K considered for label propagation, we reduce the number of annotation

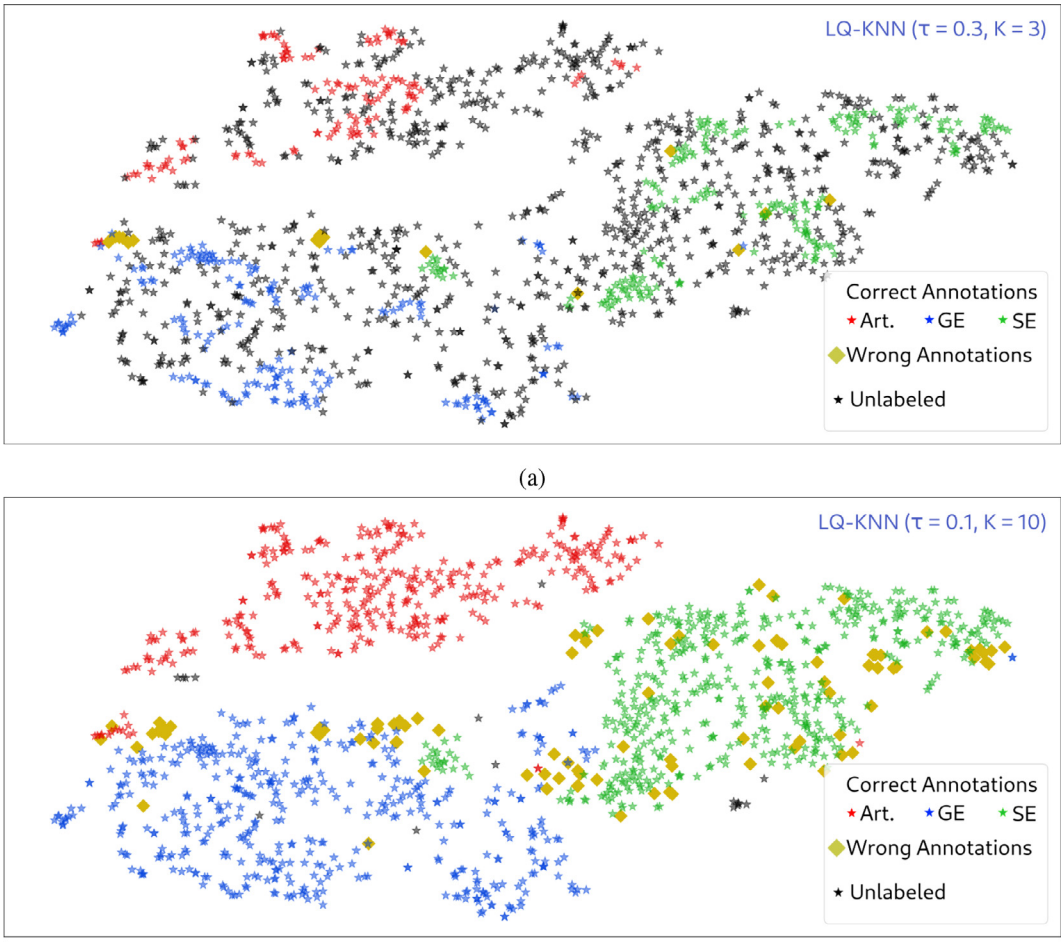


Fig. 14. Experiment 1: Label propagation for the HITS dataset using LQ-KNN with $K = 3$ and $\tau = 0.3$, and with $K = 10$ and $\tau = 0.1$. (a) LQ-KNN results with $K = 3$ and $\tau = 0.3$ (36.32% of labeled samples, with 96.44% accuracy). (b) LQ-KNN results with $K = 10$ and $\tau = 0.1$ (98.85% of labeled samples, with 92.52% accuracy). The diamonds corresponds to the wrongly labeled samples. The manually labeled samples are not shown here, for clarity. We note that the most of errors are located at the boundaries between two clusters of different classes.

errors by labeling fewer samples. Additionally, Fig. 14 also shows that most of the annotation errors are located between the boundaries of clusters of different classes; if we increase the local quality threshold, incorrectly projected samples that are at the boundaries between clusters are not going to be used for label propagation, thus reducing the number of propagated errors.

Moreover, in general, the higher the annotation accuracy, the less samples are automatically labeled, which is why when we increase τ, we have higher annotation accuracies but fewer newly labeled samples. The success of our method strongly depends on the structure assumption (Chapelle et al., 2009). The local quality criterion (Lueks et al., 2011) that we use allows it to be guaranteed that if the structure assumption is true in the feature space obtained by the auto-encoder model, then it should be true in the 2D reduced space obtained by t-SNE if the local quality criterion is verified. We introduce some flexibility to the local quality criterion by using a threshold τ that allows us to tolerate more changes in the projected space, and also allows us to label more samples. That is why the higher τ, the fewer samples we annotate, because the space of samples that can be annotated is reduced to only 'good quality' samples. Furthermore, we can see that choosing good values for K and τ is not trivial, and it depends on the application: if the quality of the labels is crucial for the application, higher values of τ should be favored (*e.g.*, $0.3 \leq \tau \leq 0.7$) and/or smaller values of K (*e.g.*, $K \leq 5$), paying with fewer labeled samples, whereas if the quality of the labels is not crucial and the number of samples is, higher values of K should be favored (*e.g.*, $K > 5$) and/or smaller values of τ (*e.g.*, $\tau \leq 0.3$). Finally, one last interesting point highlighted by this experiment is the importance of the propagation order. Indeed, in our method we propose to start labeling the samples of higher local qualities to establish an annotation order. The rationale behind this is that high local quality samples better represent the local space where they are located than lower quality samples, so it is more likely that the samples located in that zone have the same label as the highest local quality samples, rather than the lowest ones. For the results obtained, this can be seen especially for high values of K, whereas for small values, there is no important difference. This means that for applications that need to consider large neighborhoods for label propagation, the labeling order is very important.

Experiment 2

This experiment confirms the interest of automatically selecting the best 2D projection using the silhouette score. Indeed, both propagation methods achieve considerably better performances for the best selected projection than for the worse selected projection. As our method relies on the structure assumption, it is important to try to keep the high-dimensional structures in the lower-dimensional space. The silhouette score selection strategy (*i.e.*, highest score) does this by selecting the projection that al-

lows the samples with the same label in the same structure to be kept. Moreover, this experiment shows an advantage of LQ-KNN with respect to Std-KNN. Indeed, LQ-KNN is more cautious when propagating the labels, as it labels fewer samples but with higher accuracy. This result is interesting, as it shows that LQ-KNN is more robust than Std-KNN against bad 2D projections.

Experiment 3

This experiment confirms that our method improves the classification performances. On the one hand, for the three tested datasets, we observe that training a CNN on the dataset obtained with LQ-KNN (*i.e.*, $K = 10$, $\tau = 0.1$) gives better results than training a CNN on a dataset obtained without label propagation or with Std-KNN label propagation. Additionally, this experiment also confirms that using robust loss functions is beneficial when using automatically labeled data, as they allow annotation error to be compensated for. This also explain why, when using robust loss functions, Std-KNN and LQ-KNN have similar performances. Indeed, the noise in the labels introduced by both methods is similar (for the chosen parameters LQ-KNN is slightly better) so the robust loss function allows to compensate this difference. However, when the annotation error difference increases between the two label propagation methods, the robust loss function does not allow to compensate this gap, giving better classification performances to LQ-KNN datasets than to Std-KNN datasets, as the OrganCMNIST results showed it. On the other hand, this experiment also confirms the interest in using label propagation methods to automatically annotate data and to increase the test performances of the models developed for a real medical dataset. Label propagation allows the performances of a CNN to be increased, to give better results than a CNN trained on a limited dataset. When we start by propagating the labels to less than 50% of the available labeled samples, LQ-KNN outperforms Std-KNN. When the proportion of labeled samples increases, both LQ-KNN and Std-KNN have similar global behaviors, even if the LQ-KNN datasets give classifiers with better mean performances, specially when using robust loss functions.

Experiment 4

This experiment shows the stability of our method for a large-scale dataset, and the benefits of LQ-KNN propagation to improve the final classification performances of a model. Models trained on the LQ-KNN-Large dataset outperformed all of the Std-KNN trained models for both loss functions (including those of experiment 3). Additionally, by using a robust loss function to compensate for the introduced label noise, we were able to outperform HITS Whole CE/GCE in terms of MCC, gaseous emboli and solid emboli accuracies. However, this comes at the expense of a decrease in the artifact accuracy. Our hypothesis that, in this larger-scale dataset, our 2D representations of the original HITS do not verify the structure assumption anymore (this might be because in the original high-dimensional space the structure hypothesis is not verified neither or because our auto-encoder model is not adapted to this new dataset). Due to this, when we automatically propagate the labels from the labeled samples to the unlabeled samples, we introduce an unexpectedly high noise in the labels of the artifact class that the robust loss functions cannot compensate for. However, as the results show, our LQ-KNN label propagation method is more stable, as its artifact accuracy is higher than that obtained with Std-KNN, and its gaseous emboli accuracy remains similar (or even better) to the HITS Whole gaseous emboli accuracy.

Choice of k_s and k_t

We discuss the choice of $k_s = 10$ and $k_t = 10$ for all of the experiments. These parameters have an influence on the computation of the global and local quality measures of the projection; k_s controls the size of the neighborhood that we use to evaluate the structure preservation during the projection, while k_t controls the errors in terms of the rank change. As shown in Lueks et al. (2011), the global quality varies smoothly for increasing values of k_s and k_t, and between consecutive values of k_s and k_t there is little variation of the global quality. For high values of k_s and k_t we have high values of global quality, as we tolerate more errors and we consider wider neighborhoods to compute the quality. However, a high value of the global quality by itself does not necessarily mean that, globally, the neighborhood of the samples was well preserved during the projection step. Indeed, the value of the global quality should be interpreted with the values of k_s and k_t: the smaller k_s and k_t, the more the neighborhood structure of the samples is preserved during projection, which is why we have smaller global quality values (dimensionality reduction always modifies the neighborhood of samples, as it reduces the number of degrees of freedom). We choose $k_s = 10$ and $k_t = 10$, as we propagate labels from labeled samples to their unlabeled neighbors, so it is important to have a meaningful value of the global and local qualities to select the samples that can benefit from label propagation (which means that we prefer smaller values of k_s and k_t).

Limitations

Our approach has several limitations. First, the validation of the framework was limited to 3 datasets, and one type of classifier. Evaluating our method on more types of data (not only images) and models would help to better study its genericity and effectiveness. Secondly, a simple feature extraction model (AE) was used, and the influence of different types of models has not been quantitatively measured. Thirdly, our optimal selection strategy can be expensive to compute and only takes advantage of the labeled samples, which can lead to sub-optimal projections for label propagation. Fourthly, even if we show that the choice $k_s = 10$, $k_t = 10$, $K = 10$ and $\tau = 0.1$ tend to give good results for different datasets, more efforts need to be done to propose an easier strategy for hyper-parameter selection. Finally, for classification, only one robust loss function was tested, but other loss functions can be used such as symmetric CE (Wang et al., 2019) and other strategies can be adopted to deal with noisy-labels (Song et al., 2021).

5. Conclusions

We proposed a semi-supervised learning approach for semi-automatic data annotation from sparsely annotated datasets with controlled annotation errors. To do this, we start by extracting features from the data in an unsupervised manner, using an auto-encoder. Then, we use t-SNE as the dimensionality reduction technique to project the learned representations of the auto-encoder onto a 2D space, and we select the best projection using the silhouette score. Then, we propagate the labels from the few labeled samples to some unlabeled samples using a local quality criterion based on (Lueks et al., 2011). This criterion allows the labels from labeled samples to be propagated to their neighbors only if both samples (*i.e.*, labeled and unlabeled) are close not only in the 2D space, but also in the higher-dimensional space learned by the auto-encoder. Finally, to compensate for the errors made by the label propagation method, we use a robust loss function for classification, the GCE loss.

Our experiments show several results. First, our label propagation method outperforms state-of-the-art methods such as OPF-semi. Secondly, the choice of the hyper-parameters of our proposed method allows us to control the annotation errors. Thirdly, we show that the combination of our label propagation method with robust loss functions improves the final classification perfor-

mances of the trained models on the semi-automatically labeled datasets obtained. Fourthly, our method allows similar (and even better) classification performances to be achieved than those obtained using a fully manually labeled dataset. This last point is particularly interesting, as our method takes less than 0.2 ms to annotate one sample with high accuracy, compared to 8 s for a human expert (HITS dataset). Finally, we showed that our method is applicable to different datasets by evaluating it on three different datasets (two of which are publicly available). The two blocks that need to be adapted to each dataset are the feature extraction block and the classification block.

As perspectives, we would like to incorporate a self-training (Rosenberg et al., 2005) strategy into our method, using robust loss functions and Bayesian approaches. Moreover, during the self-training step, we would like to directly learn the 2D representation used to propagate the labels, instead of using t-SNE, which would allow us to avoid the expensive search of the optimal projection space, one of the main weaknesses of our approach. Furthermore, as our method allows manual annotation of any samples, we would like to proceed as (Benato et al., 2021), to manually annotate the samples that our method cannot label with high confidence. Finally, as our main goal is to improve the classification performances of TCD data, we would like to incorporate the use of audio signals, to improve the learned representations (autoencoder and 2D space) and the classification performances.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Yamil Vindas: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing – original draft, Visualization. **Blaise Kévin Guépié:** Conceptualization, Validation, Writing – review & editing, Visualization. **Marilys Almar:** Conceptualization, Investigation, Resources, Data curation, Writing – review & editing. **Emmanuel Roux:** Conceptualization, Methodology, Software, Validation, Formal analysis, Writing – review & editing, Visualization. **Philippe Delachartre:** Conceptualization, Validation, Resources, Visualization, Supervision, Project administration, Funding acquisition.

Acknowledgment

This work was carried out in the context of the CAREMB project funded by the Auvergne-Rhône-Alpes region, within the *Pack Ambition Recherche* program. This work was performed within the framework of theLABEX CELYA (ANR-10-LABX-0060) and PRIMES (ANR-11-LABX-0063) of Universite de Lyon, within the program "Investissements d'Avenir" (ANR-11-IDEX-0007) operated by the-French National Research Agency (ANR).

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.media.2022.102437.

References

Aggarwal, S.K., Delahunty RN, N., Menezes, L.J., Perry, R., Wong, B., Reinthaler, M., Ozkor, M., Mullen, M.J., 2018. Patterns of solid particle embolization during transcatheter aortic valve implantation and correlation with aortic valve calcification. J Interv Cardiol 31 (5), 648–654. doi:10.1111/joic.12526. Number: 5.

Amorim, W.P., Falcão, A., Carvalho, M.H., 2014. Semi-supervised pattern classification using optimum-path forest. In: 2014 27th SIBGRAPI Conference on Graphics, Patterns and Images, pp. 111–118.

Belkin, M., Niyogi, P., Sindhwani, V., 2006. Manifold regularization: a geometric framework for learning from labeled and unlabeled examples. Journal of Machine Learning Research 7 (85), 2399–2434. URL: http://jmlr.org/papers/v7/belkin06a.html.

Benato, B.C., Gomes, J.F., Telea, A.C., Falcão, A.X., 2021. Semi-automatic data annotation guided by feature space projection. Pattern Recognit 109, 107612. doi:10.1016/j.patcog.2020.107612. URL: https://linkinghub.elsevier.com/retrieve/pii/S0031320320304155.

Benato, B.C., Telea, A.C., Falcao, A.X., 2018. Semi-supervised learning with interactive label propagation guided by feature space projections. In: 2018 31st SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI). IEEE, pp. 392–399. doi:10.1109/SIBGRAPI.2018.00057. URL: https://ieeexplore.ieee.org/document/8614354/.

Bilic, P., Christ, P.F., et al., 2019. The liver tumor segmentation benchmark (lits) CoRR arXiv:1901.04056.

Chapelle, O., Scholkopf, B., Zien, E.A., 2009. Semi-supervised learning (Chapelle, O. et al., Eds.; 2006). IEEE Trans. Neural Networks 20 (3). doi:10.1109/TNN.2009.2015974. pp. 542–542, Conference Name: IEEE Transactions on Neural Networks.

Chen, M., Shi, X., Zhang, Y., Wu, D., Guizani, M., 2017. Deep features learning for medical image analysis with convolutional autoencoder neural network. IEEE Trans. Big Data doi:10.1109/TBDATA.2017.2717439. pp. 1–1, URL: http://ieeexplore.ieee.org/document/7954012/.

de Rosa, G.H., Papa, J.P., 2021. Opfython: a python implementation for optimum-path forest. Software Impacts 100113. doi:10.1016/j.simpa.2021.100113.

Doersch, C., Gupta, A., Efros, A.A., 2015. Unsupervised visual representation learning by context prediction. In: 2015 IEEE International Conference on Computer Vision (ICCV), pp. 1422–1430.

Gencer, M., Bilgin, G., Aydin, N., 2013. Embolic doppler ultrasound signal detection via fractional fourier transform. In: 2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). IEEE, pp. 3050–3053. doi:10.1109/EMBC.2013.6610184.

Goldberger, J., Ben-Reuven, E., 2017. Training deep neural-networks using a noise adaptation layer. ICLR.

Guepie, B.K., Martin, M., Lacrosaz, V., Almar, M., Guibert, B., Delachartre, P., 2019. Sequential emboli detection from ultrasound outpatient data. IEEE J Biomed Health Inform 23 (1), 334–341. doi:10.1109/JBHI.2018.2808413. Number: 1, URL: https://ieeexplore.ieee.org/document/8300318/.

Guépié, B.K., Sciolla, B., Millioz, F., Almar, M., Delachartre, P., 2017. Discrimination between emboli and artifacts for outpatient transcranial doppler ultrasound data. Medical & Biological Engineering & Computing 55 (10), 1787–1797. doi:10.1007/s11517-017-1624-z. Number: 10.

Hicks, S.A., Strümke, I., Thambawita, V., Hammou, M., Riegler, M.A., Halvorsen, P., Parasa, S., 2021. On evaluation metrics for medical applications of artificial intelligence. medRxiv doi:10.1101/2021.04.07.21254975.

Johnson, W., Onuma, O., Owolabi, M., Sachdev, S., 2016. Stroke: a global response is needed. Bull. World Health Organ. 94 (9). doi:10.2471/BLT.16.181636. 634–634A, Number: 9.

Jolliffe, I.T., Cadima, J., 2016. Principal component analysis: a review and recent developments. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 374 (2065), 20150202. doi:10.1098/rsta.2015.0202. Number: 2065.

Karahoca, A., Tunga, M.A., 2015. A polynomial based algorithm for detection of embolism. Soft comput 19 (1), 167–177. doi:10.1007/s00500-014-1240-x. Number: 1.

Kingma, D.P., Rezende, D.J., Mohamed, S., Welling, M., 2014. Semi-supervised learning with deep generative models. In: Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2. MIT Press, Cambridge, MA, USA, pp. 3581–3589.

LeCun, Y., Cortes, C., 2010. MNIST handwritten digit databaseURL: http://yann.lecun.com/exdb/mnist/.

Lee, J.A., Verleysen, M., 2009. Quality assessment of dimensionality reduction: rank-based criteria. Neurocomputing 72 (7), 1431–1443. doi:10.1016/j.neucom.2008.12.017. Number: 7–9, URL: https://linkinghub.elsevier.com/retrieve/pii/S0925231209000101.

Lueks, W., Mokbel, B., Biehl, M., Hammer, B., 2011. How to evaluate dimensionality reduction? - improving the co-ranking matrix arXiv:1110.3917 [cs].

Maaten, L.v.d., Hinton, G., 2008. Visualizing data using t-SNE. Journal of Machine Learning Research 9 (86), 2579–2605. Number: 86, URL: http://jmlr.org/papers/v9/vandermaaten08a.html.

Markus, H., Israel, D., Brown, M., Loh, A., Buckenham, T., Clifton, A., 1993. Microscopic air embolism during cerebral angiography and strategies for its avoidance. The Lancet 341 (8848), 784–787. doi:10.1016/0140-6736(93)90561-T. Number: 8848.

Markus, H.S., Punter, M., 2005. Can transcranial doppler discriminate between solid and gaseous microemboli?: assessment of a dual-frequency transducer system. Stroke 36 (8), 1731–1734. doi:10.1161/01.STR.0000173399.20127.b3. Number: 8.

McInnes, L., Healy, J., Melville, J., 2020. UMAP: Uniform manifold approximation and projection for dimension reduction arXiv:1802.03426 [cs, stat].

Packer, E., Bak, P., Nikkila, M., Polishchuk, V., Ship, H.J., 2013. Visual analytics for spatial clustering: using a heuristic approach for guided exploration. IEEE Trans Vis Comput Graph 19 (12), 2179–2188. doi:10.1109/TVCG.2013.224. Number: 12, URL: http://ieeexplore.ieee.org/document/6634158/.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S., 2019. Pytorch: an imperative style, high-performance deep learning library. In: Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R. (Eds.), Advances in Neural Information Processing Systems 32. Curran Associates, Inc., pp. 8024–8035. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E., 2011. Scikit-learn: machine learning in Python. Journal of Machine Learning Research 12, 2825–2830.

Pereyra, G., Tucker, G., Chorowski, J., Kaiser, L., Hinton, G.E., 2017. Regularizing neural networks by penalizing confident output distributions. In: 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24–26, 2017, Workshop Track Proceedings. OpenReview.net. URL: https://openreview.net/forum?id=HyhbYrGYe.

Rosenberg, C., Hebert, M., Schneiderman, H., 2005. Semi-supervised self-training of object detection models. In: 2005 Seventh IEEE Workshops on Applications of Computer Vision (WACV/MOTION'05) - Volume 1. IEEE, pp. 29–36. doi:10.1109/ACVMOT.2005.107. URL: http://ieeexplore.ieee.org/document/4129456/.

Rosenkranz, M., Fiehler, J., Niesen, W., Waiblinger, C., Eckert, B., Wittkugel, O., Kucinski, T., Röther, J., Zeumer, H., Weiller, C., Sliwka, U., 2006. The amount of solid cerebral microemboli during carotid stenting does not relate to the frequency of silent ischemic lesions. American Journal of Neuroradiology 27 (1), 157–161. Number: 1 Publisher: American Journal of Neuroradiology Section: INTERVENTIONAL, URL: http://www.ajnr.org/content/27/1/157.

Rousseeuw, P.J., 1987. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. J Comput Appl Math 20, 53–65. doi:10.1016/0377-0427(87)90125-7. URL: https://linkinghub.elsevier.com/retrieve/pii/0377042787901257.

Serbes, G., Aydin, N., 2014. Denoising performance of modified dual-tree complex wavelet transform for processing quadrature embolic doppler signals. Medical & Biological Engineering & Computing 52 (1), 29–43. doi:10.1007/s11517-013-1114-x. Number: 1.

Serena, J., Jimenez-Nieto, M., Silva, Y., Castellanos, M., 2010. Patent foramen ovale in cerebral infarction. Curr Cardiol Rev 6 (3), 162–174. doi:10.2174/157340310791658794. Number: 3.

Sindhwani, V., Niyogi, P., Belkin, M., 2005. Beyond the point cloud: from transductive to semi-supervised learning. In: Proceedings of the 22nd International Conference on Machine Learning.

Sombune, P., Phienphanich, P., Muengtaweepongsa, S., Ruamthanthong, A., Tantibundhit, C., 2016. Automated embolic signal detection using adaptive gain control and classification using ANFIS. In: 2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). IEEE, pp. 3825–3828. doi:10.1109/EMBC.2016.7591562.

Sombune, P., Phienphanich, P., Phuechpanpaisal, S., Muengtaweepongsa, S., Ruamthanthong, A., Tantibundhit, C., 2017. Automated embolic signal detection using deep convolutional neural network. In: 2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). IEEE, pp. 3365–3368. doi:10.1109/EMBC.2017.8037577.

Song, H., Kim, M., Lee, J.-G., 2019. Selfie: Refurbishing unclean samples for robust deep learning. ICML.

Song, H., Kim, M., Park, D., Shin, Y., Lee, J.-G., 2021. Learning from noisy labels with deep neural networks: A survey. Unpublished.

Spencer, M.P., Ackerstaff, R.G., Babikian, V.L., Georgiadis, D., Russell, D., Siebler, M., Stump, D., 1995. Basic identification criteria of doppler microembolic signals. Stroke 26 (6). doi:10.1161/01.STR.26.6.1123. 1123–1123, Number: 6.

Tafsast, A., Ferroudji, K., Hadjili, M.L., Bouakaz, A., Benoudjit, N., 2018. Automatic microemboli characterization using convolutional neural networks and radio frequency signals. In: 2018 International Conference on Communications and Electrical Engineering (ICCEE). IEEE, pp. 1–4. doi:10.1109/CCEE.2018.8634521.

Tenenbaum, J.B., 2000. A global geometric framework for nonlinear dimensionality reduction. Science 290 (5500), 2319–2323. doi:10.1126/science.290.5500.2319. Number: 5500.

Tschannen, M., Bachem, O., Lucic, M., 2018. Recent advances in autoencoder-based representation learning arXiv:1812.05069 [cs, stat].

Vindas, Y., Roux, E., Guépié, B.K., Almar, M., Delachartre, P., 2021. Semi-supervised annotation of transcranial doppler ultrasound micro-embolic data. In: 2021 IEEE International Ultrasonics Symposium (IUS), pp. 1–4. doi:10.1109/IUS52206.2021.9593847.

Wallace, S., Døhlen, G., Holmstrøm, H., Lund, C., Russell, D., 2015. Cerebral microemboli detection and differentiation during transcatheter closure of atrial septal defect in a paediatric population. Cardiol Young 25 (2), 237–244. doi:10.1017/S1047951113002072. Number: 2, URL: https://www.cambridge.org/core/product/identifier/S1047951113002072/type/journal_article.

Wang, Y., Ma, X., Chen, Z., Luo, Y., Yi, J., Bailey, J., 2019. Symmetric cross entropy for robust learning with noisy labels. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 322–330.

Weston, J., Ratle, F., Collobert, R., 2008. Deep learning via semi-supervised embedding. ICML '08.

Xiao, T., Xia, T., Yang, Y., Huang, C., Wang, X., 2015. Learning from massive noisy labeled data for image classification. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, pp. 2691–2699. doi:10.1109/CVPR.2015.7298885.

Yan, Y., Xu, Z., Tsang, I.W., Long, G., Yang, Y., 2016. Robust semi-supervised learning through label aggregation. In: AAAI, pp. 2244–2250. URL: http://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/view/12312.

Yang, J., Shi, R., Wei, D., Liu, Z., Zhao, L., Ke, B., Pfister, H., Ni, B.. Medmnist v2: a large-scale lightweight benchmark for 2D and 3D biomedical image classification arXiv preprint.

Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O., 2018. The unreasonable effectiveness of deep features as a perceptual metric. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. IEEE, pp. 586–595. doi:10.1109/CVPR.2018.00068. URL: https://ieeexplore.ieee.org/document/8578166/.

Zhang, Z., Sabuncu, M.R., 2018. Generalized cross entropy loss for training deep neural networks with noisy labels. In: Bengio, S., Wallach, H.M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R. (Eds.), Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3–8, 2018, Montréal, Canada, pp. 8792–8802.

Zhou, C., Paffenroth, R.C., 2017. Anomaly detection with robust deep autoencoders. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, pp. 665–674. doi:10.1145/3097983.3098052.

Zhu, X., Ghahramani, Z., 2002. Learning from labeled and unlabeled data with label propagation. Technical Report CMU-CALD-02-107. School of Computer Science, Carnegie Mellon University, Pittsburgh, PA.