LABEL PROPAGATION ADAPTIVE RESONANCE THEORY FOR SEMI-SUPERVISED CONTINUOUS LEARNING

Taehyeong Kim\textsuperscript{1,2}, Injune Hwang\textsuperscript{1}, Gi-Cheon Kang\textsuperscript{1}, Won-Seok Choi\textsuperscript{1}, Hyunseo Kim\textsuperscript{1}, Byoung-Tak Zhang\textsuperscript{1}

\textsuperscript{1}Seoul National University, Republic of Korea
\textsuperscript{2}AI Lab, CTO Division, LG Electronics, Republic of Korea
{\texttt{thkim, ijhwang, gckang, wchoi, hskim, btzhang}@bi.snu.ac.kr}

ABSTRACT
Semi-supervised learning and continuous learning are fundamental algorithms for human-level intelligence. To deal with real-world problems where labels are rarely given and the opportunity to access the same data is limited, it is necessary to apply these two algorithms in a joined fashion. In this paper, we propose Label Propagation Adaptive Resonance Theory (LPART) for semi-supervised continuous learning. LPART uses label propagation mechanism to perform classification and gradually improves accuracy as the observed data accumulates. We evaluated our proposed model on both visual (SVHN, MNIST, CIFAR-10) and audio (NSynth) datasets by adjusting the ratio of the labeled and unlabeled data. The accuracies are much higher when both labeled and unlabeled data are used, demonstrating that LPART takes significant advantage in environments where data label is scarce.

Index Terms— Semi-supervised learning, continuous learning, adaptive resonance theory, label propagation

1. INTRODUCTION
Over the last few years, the deep neural network models have shown remarkable progresses especially in visual object recognition, speech recognition, and autonomous robot control. However, the use of deep learning has a major practical shortcoming that it requires time and labor not only to collect massive amount of data, but to label them. In this aspect, the fields of semi-supervised learning, continuous learning, transfer learning, and meta-learning are in the spotlight.

The semi-supervised learning algorithm \cite{1, 2} tackles a problem in which the unlabeled data is abundant but the labeled data is extremely limited. The continuous learning algorithm \cite{3, 4} aims to learn without catastrophic forgetting the former knowledge from streams of data, allowing the model adapt to the ever-changing environments. Although each algorithm is promising on its own, we argue that both algorithms should be applied in a joined fashion to deal with many real-world environments, where labels of data are rarely given and the data once learned cannot be accessed again. Further technique such as label propagation can be added to cope with this difficulty.

Label propagation, a mechanism for semi-supervised learning, is a method of inferring a class of unlabeled cluster with help of labeled ones \cite{5}. It is assumed that the clusters close to each other in the feature space tend to belong to similar class. Information from the labeled cluster is repeatedly propagated to dense, unlabeled data regions. Recently, various methods for label propagation \cite{6, 7} have been proposed.

In this study, we propose Label Propagation Adaptive Resonance Theory (LPART) for semi-supervised continuous learning. It employs the label propagation mechanism to the ART network \cite{8}. Specifically, we used two-fold learning process: (1) feature extraction using variational autoencoders (VAE) \cite{9} and (2) clustering of the extracted features and inference using LPART. In the first fold, we train VAE in a weakly-supervised manner by using pair loss. Then, LPART takes the features as input, and learn to infer the class of the unlabeled node by leveraging the label propagation mechanism. However, the inference for the unlabeled node could be woefully inaccurate when the number of observed nodes is small. So, we propose the metrics for measuring two different uncertainty scores. By consulting these uncertainty scores, LPART could defer the classification for the node with high uncertainties. We expect our proposed algorithm to be capable of learning continuously without catastrophic forgetting, even in the rarely labeled environments.

2. ADAPTIVE RESONANCE THEORY
Adaptive Resonance Theory (ART) is a self-organizing neural network inspired by the brain information processing mechanisms \cite{8}. ART uses the interaction of ‘top-down’ expectation and ‘bottom-up’ sensory information to learn adaptively, using resonance. There are two general principles in ART: (1) the knowledge is strengthened when the sensation is strong enough and the expectation matches well with the sensation,
and (2) a new knowledge is learned when the sensation is recognized but the matches between all available expectations and the sensation is below the criterion. In terms of being conservative while learning new, the ART system can be a solution for the continuous learning problems.

Various ART networks such as Fuzzy ART [10], ARTMAP [11], and Fuzzy ARTMAP (FAM) [12]. Fuzzy ART can process real-valued data by using fuzzy set theory. ARTMAP uses supervised learning and classification system that is built up from a pair of ART module. FAM integrates the advantages of both Fuzzy ART and ARTMAP.

There are other ART networks for learning in a semi-supervised manner, such as semi-supervised Bayesian ARTMAP (SSBA) [13] and semi-supervised Fuzzy ARTMAP (ss-FAM) [14]. SSBA employs EM algorithm based on Bayesian ARTMAP (BA) [15] to adjust its parameter, which realizes the soft assignment of training samples instead of winner-take-all strategy. ssFAM relies on FAM but adopts a tunable network parameter called category prediction error tolerance in the map field, which achieves semi-supervised learning.

The present study differs from previous studies in that it applies label propagation mechanism for semi-supervised learning. In addition, the label propagation mechanism can be applied to various kinds of ART networks, which allows the extension from LPART. Also, the uncertainty measurement method proposed in this study can be used in various ways in semi-supervised continuous learning environments.

3. METHODOLOGY

3.1. Feature Extraction with VAE

It is necessary to extract features that are easy to cluster for ART to classify high-dimensional data properly. Variational autoencoder (VAE), a deep learning-based unsupervised learning method, is considered suitable to provide a way to extract useful features. However, basic VAE does not have any explicit constraints to improve clustering. In this regard, the study of learning representations using oracle triplets provides the insights needed for this study [16]. We also repurposed the VAE architecture for semi-supervised continuous learning.

In this study, we use a simplified triplet-based VAE to extract features. It uses only some dimensions \( d \) in the latent space for VAE encoder to produce class-embedded representation \( \mu_d \). Additional pair loss is introduced, which depends on whether the class of the previous sample and the current sample is the same or not. The pair loss between previous and current class-embedded representations is defined using the Euclidean distance as similarity measure.

\[
\mathcal{L}_{\text{pair}} = \begin{cases} 
\| \mu_{d,\text{prev}} - \mu_{d,\text{curr}} \|_2^2, & \text{if } y_{\text{prev}} = y_{\text{curr}} \\
-\| \mu_{d,\text{prev}} - \mu_{d,\text{curr}} \|_2^2, & \text{otherwise}
\end{cases}
\]  

Here, \( y \) denotes a label of input sample. We optimize parameters by maximizing the ELBO (evidence lower bound) [9] and minimizing the pair loss. With scaling factor \( \lambda \), the total loss to be minimized is as shown in Equation 2.

\[
\mathcal{L} = -\mathcal{L}_{\text{ELBO}} + \lambda \mathcal{L}_{\text{pair}}
\]  

3.2. The LPART Algorithm

When an input data \( x_i \) is given, we encode it using the VAE previously described to get the 0-to-1 normalized class-embedded representation as \( r_i \). As in Fuzzy ART, \( I_i \) is the complement coding of \( r_i \). For a node \( j \) with a weight vector \( w_j \), the choice function \( T_j \) and the match function \( V_j \) of \( I_i \) are defined as:

\[
T_j(I_i) = \frac{\| I_i \wedge w_j \|_1}{\alpha + \| w_j \|_1}, \quad V_j(I_i) = \frac{\| I_i \wedge w_j \|_1}{\| I_i \|_1}
\]  

where \( \wedge \) is an element-wise minimum operator and \( \alpha > 0 \) is a choice parameter which can be seen as a regularization parameter.

If the value of \( V_j(I_i) \) is greater than a vigilance parameter \( \rho \), we say that the node \( j \) has matched \( x_i \), or been activated. Among all of the activated nodes, a winner \( J \) with the highest value of \( T_j \) is selected. It can be seen as the best-fit node for the input, and we update its weight vector considering \( I_i \) with a learning rate \( \beta \) between 0 and 1 as shown in Equation 4.

\[
w_j^{\text{new}} = \beta(I_i \wedge w_j^{\text{old}}) + (1 - \beta)w_j^{\text{old}}
\]  

On the other hand, if no node matches the input, a new node is created with initial parameter set as \( I_i \). This newborn node can grow larger throughout the subsequent iterations. Creation of a node is more frequent with a larger value of \( \rho \). By manipulating the value of \( \rho \), we can balance the rigidity of node. If it gets too small, one node covers up too many inputs, making the consistency of node vague. Therefore, we use sufficiently large value for the vigilance parameter.

Another crucial part of LPART, the label propagation mechanism, will be explained in the following section. The overall LPART algorithm is described in Algorithm 1.

3.3. Label Propagation Mechanism

Label propagation is triggered when an input data activates one or more nodes. The co-activated nodes can be considered to be located in vicinity of each other in the feature space, which in turn implies a high relevance between them. It is natural, therefore, to estimate the label of a label-absent node—a node that does not contain any input with a label—as the representative value of labels of co-activated nodes. However, the numerical value itself is meaningless, so we use a distribution over all labels instead of a single value. We call this a label density function and denote by \( q_j(y) \) roughly means...
how probable a node $j$ will be in class $y$. When a new node $n$ is created, $q_n$ has a uniform distribution with small values.

Once a labeled sample is added to a node, the density of its label increases by one. For a label-absent node $k$, label density function is updated by averaging those of co-activated nodes:

$$q^\text{new}_k(y) = \left( \delta \times \frac{\sum_{j \in A \setminus \{k\}} q^\text{old}_j(y)}{\sum_{y'} \sum_{j \in A \setminus \{k\}} q^\text{old}_j(y')} \right) + \left( 1 - \delta \right) \times \frac{q^\text{old}_k(y)}{\sum_{y'} q^\text{old}_k(y')} \times \frac{1}{C}$$

where $\delta$ is a propagation rate. The reason why the sum of $q_k$ over all labels is less than one is to indicate that it is still not certain which class this node belongs to. $C > 1$ can be interpreted as a kind of uncertainty parameter, which will be further discussed in Section 3.4.

Finally, the probability distribution of labels for each node is easily obtained by normalizing the label density function:

$$p_j(y) = \frac{q_j(y)}{\sum_{y'} q_j(y')}$$

and we can infer a class of input data by first finding a winning node and select a label with the highest probability.

### 3.4. Measurement of Uncertainty

We propose two different metrics to measure the uncertainty of the classification results. The first uncertainty, $u_1(x_i)$, is measured by the entropy of the classification probability, as shown in Equation 7. This method measures how evenly distributed the categories of labeled data learned by each node that can be used to identify high-impurity nodes.

$$u_1(x_i) = - \sum_y p_{J(x_i)}(y) \log p_{J(x_i)}(y)$$

$J(x_i)$ is a winning node with given input $x_i$ and $p_{J(x_i)}(y)$ is the probability that input data $x_i$ belongs to class $y$.

The second uncertainty $u_2(x_i)$ is based on the number of labeled input data in each node as shown in Equation 8. It allows them to filter out highly unreliable recognition results from the datasets with insufficient number of labels.

$$u_2(x_i) = 1 - tanh(c \cdot \sum_y q_{J(x_i)}(y))$$

Here, $c$ is a constant for sensitivity. The combination of these two uncertainties comes in handy for real-world problems; for example, we can withhold judgments on samples with high uncertainties during continuous learning.

## 4. EXPERIMENTS

### 4.1. Dataset

We experiment with our proposed model on MNIST [17], SVHN [18], CIFAR-10 [19], and NSynth [20] datasets. The MNIST and SVHN datasets consist of the digit images with class labels ranging from 0 to 9. The CIFAR-10 consists of 32x32 color images in 10 classes. The NSynth dataset contains 306,043 four-second audio recordings from 1,006 instruments. Each recording is labeled with one of the eleven high-level groups. Note that we omit the synth lead class label for less number of samples and only use validation and test split in the NSynth dataset.

![Fig. 1. The vector visualization of feature extraction results. (b) and (c) used t-SNE [21] for dimension reduction.](image)

### 4.2. Semi-supervised Learning

In this section, we formally describe the experimental details for the semi-supervised learning on the MNIST, SVHN, CIFAR-10, and NSynth datasets. As shown in Table 1, we verify the semi-supervised classification performance by adjusting the ratio of the labeled and unlabeled data. Also, we compare our proposed model with the FAM. Because the FAM model is based on fully-supervised learning, we could not report the performance of the FAM on semi-supervised setting. We report the average performance of thirty trials. Each trial is trained by one epoch.
4.3. Semi-supervised Continuous Learning

In this experiment, we expand our experiment to semi-supervised continuous learning on NSynth dataset. As shown in Figure 2, we perform two experiments: (1) comparison between LPART with FAM ranging from 1 to 20 epochs, and (2) the uncertainty rate and classification accuracy by epochs. We also average the performance of ten trials.

5. RESULTS AND DISCUSSION

5.1. Semi-supervised Learning

Figure 1 visualizes the feature extraction results of the datasets used in this study. The classification accuracies with various few-label settings are summarized in Table 1. In all experimental setups, the best results were obtained from our model using both labeled and unlabeled data. The accuracy is much higher than when only the labeled data is used. In particular, when only the labeled data is used, the performance drops down as the data decreases. However, when the unlabeled data is used together, the decrease margin is not significant. In some datasets, such as CIFAR-10, the classification performance is not good. This is because the suitability of the extracted feature has affected performance. However, from the standpoint of semi-supervised learning performance evaluation, the unlabeled data has a positive effect on performance improvement. When proper feature extraction methods are used together, better results will be obtained.

5.2. Semi-supervised Continuous Learning

The results of semi-supervised continuous learning using the NSynth dataset are shown in Figure 2. When unlabeled data is used together with the LPART, the classification performance according to epochs rapidly increases to 90% and converges (Figure 2-a). However, when only a part of the labeled data is used, the classification performance increases slowly and is similar to the FAM. It confirms that the proposed method for semi-supervised continuous learning works properly.

We also used the uncertainty measurement method described in Section 3.4 to exclude uncertain classification results. Thresholds for two uncertainties were set, and the results of the reliable classification were filtered (Figure 2-b). As learning progresses, the number of uncertain samples continue to decrease, and reliable classification results always maintained high performance. This method is useful for applications where classification errors are fatal, and it allows us to use only reliable results in situations where data collection is scarce.

6. CONCLUSIONS

In the present study, we propose a novel approach for semi-supervised continuous learning. We applied the label propagation mechanism to the ART network and evaluated it with various datasets and experimental settings to demonstrate its effectiveness. Uncertainty measures can also be used in a variety of ways, including active learning, in real-world problems. The limitation of this study is that a pre-trained feature extractor should be used, and the suitability of the extracted feature can affect the overall performance. Future works need to deal with end-to-end training the entire system by applying an online learning method to the feature extractor.

Table 1. Classification accuracy of our model (LPART) compared to Fuzzy ARTMAP trained for a single epoch with various probabilities of the labeled data. The mean and standard deviation are drawn from 30 trials for each experiment. (unit : %)

<table>
<thead>
<tr>
<th>Dataset rate (labeled, unlabeled)</th>
<th>MNIST ($\rho = 0.99$) FAM</th>
<th>LNART</th>
<th>SVHN ($\rho = 0.98$) FAM</th>
<th>LNART</th>
<th>CIFAR-10 ($\rho = 0.95$) FAM</th>
<th>LNART</th>
<th>NSynth ($\rho = 0.95$) FAM</th>
<th>LNART</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1%, not used</td>
<td>46.2±3.0</td>
<td>46.5±3.4</td>
<td>48.1±2.8</td>
<td>48.8±2.9</td>
<td>33.0±2.5</td>
<td>30.9±2.9</td>
<td>46.8±9.3</td>
<td>48.3±10.9</td>
</tr>
<tr>
<td>0.1%, 99.9%</td>
<td>-</td>
<td>94.2±1.0</td>
<td>-</td>
<td>73.7±1.3</td>
<td>-</td>
<td>39.5±2.0</td>
<td>-</td>
<td>63.1±1.7</td>
</tr>
<tr>
<td>0.5%, not used</td>
<td>71.8±1.9</td>
<td>70.9±1.7</td>
<td>60.5±2.0</td>
<td>59.4±1.9</td>
<td>38.8±1.2</td>
<td>37.1±1.5</td>
<td>67.2±4.3</td>
<td>68.3±4.5</td>
</tr>
<tr>
<td>0.5%, 99.5%</td>
<td>-</td>
<td>95.0±0.3</td>
<td>-</td>
<td>74.5±0.4</td>
<td>-</td>
<td>42.4±0.9</td>
<td>-</td>
<td>85.2±2.1</td>
</tr>
<tr>
<td>1.0%, not used</td>
<td>79.5±1.6</td>
<td>79.8±1.6</td>
<td>63.6±1.7</td>
<td>63.4±1.3</td>
<td>40.3±0.7</td>
<td>38.6±1.3</td>
<td>73.6±3.3</td>
<td>74.0±4.0</td>
</tr>
<tr>
<td>1.0%, 99.0%</td>
<td>-</td>
<td>95.3±0.3</td>
<td>-</td>
<td>74.8±0.4</td>
<td>-</td>
<td>42.8±0.7</td>
<td>-</td>
<td>87.6±1.6</td>
</tr>
<tr>
<td>5.0%, not used</td>
<td>90.2±0.8</td>
<td>90.1±0.8</td>
<td>70.6±0.8</td>
<td>70.8±0.6</td>
<td>41.9±0.6</td>
<td>42.6±0.9</td>
<td>84.3±1.8</td>
<td>84.4±1.4</td>
</tr>
<tr>
<td>5.0%, 95.0%</td>
<td>-</td>
<td>96.2±0.2</td>
<td>-</td>
<td>76.0±0.4</td>
<td>-</td>
<td>44.1±0.5</td>
<td>-</td>
<td>90.0±0.7</td>
</tr>
</tbody>
</table>
7. REFERENCES


