Visualization of Learning Process in Feature Space

Tomohiro Inoue, Noboru Murata, Taiki Sugiura
Waseda University
tomo.i@asagi.waseda.jp, noboru.murata@eb.waseda.ac.jp, taiki_sugiura@toki.waseda.jp

Abstract

In machine learning, the structure of feature space is an important factor that determines the performance of a model. Therefore, we can deepen our understanding of learning algorithms if we can visualize changes in the structure of feature space during the learning process. However, visualizing such changes is difficult because it requires dimensionality reduction while maintaining consistency with the data structure in high-dimensional space and in the temporal direction. In this study, we visualized feature changes during the learning process by capturing them as changes in the positional relationship between target features and time-invariant reference coordinates with a log-bilinear model.

Introduction

Machine learning tasks often use data as their input which are difficult to handle directly as numerical values, such as images and texts. Deep learning models can take such unstructured data as input and extract structured high-dimensional representations for given tasks. The extracted representations are called *features*, and the structure of feature space is an important factor in determining the performance of the model. Therefore, visualization and analysis of feature space are useful to understand the learning and prediction mechanisms of machine learning models. By visualizing the distribution and cluster structure of data in feature space, it is possible to visually evaluate learning algorithms.

The structure of feature space changes as the parameters of feature extractor change during the learning process. Visualizing this structural change can help deepen our understanding of learning algorithms. For example, in multi-task learning, where multiple tasks are used alternately, it may be possible to visually understand the impact of each task on the formation of clusters in the learning process. However, there are several problems with visualizing feature space during the learning process. First, since features are generally high-dimensional, it is necessary to reduce the dimension to a lower dimension while preserving the positional relationship among the data in the original high-dimensional space in order to visualize them. In addition, it is necessary to reduce the dimensionality so that the location of each data point in the low-dimensional space is consistent in the time direction as well as in the positional relationship among the data points at each time.

In this study, we introduce time-invariant reference coordinates to reduce feature dimensions with consistency in the time direction. This method captures temporal changes in the target features as those of the relationship between the targets and the references. The contributions of this research are as follows:

- proposal of a method to visualize the learning process in feature space,
- proposal of a method to decompose the relationship between target features into the time-varying distribution of targets and the time-invariant distribution of references,
- demonstration of visualization analysis of artificial data and real data.

Related Work

Dimentionality Reduction

To visualize high-dimensional data, it is necessary to map the data to a low-dimensional space, such as three dimensions or less, while preserving the structure in the original high-dimensional space as much as possible. Dimensionality reduction is a method to achieve this task and can be formulated as follows. Let $\mathcal{O} = \{o_1, \ldots, o_n\}$ be a set of objects to be mapped, and let $\mathcal{X} = \{\mathbf{x}_i | i = 1, \ldots, n, \mathbf{x}_i \in \mathbf{R}^d\}$ be a set of high-dimensional coordinates corresponding to each of $\mathcal{O}$. In dimensionality reduction, $\mathcal{X}$ is mapped to the low-dimensional space $\mathcal{Y} = \{\mathbf{y}_i | i = 1, \ldots, n,\ \mathbf{y}_i \in \mathbf{R}^{d'}, d' < d\}$ by mapping $f : \mathcal{X} \to \mathcal{Y}$.

A popular method of dimensionality reduction is t-SNE (van der Maaten and Hinton 2008). t-SNE is a method that preserves the neighborhood structure of points based on the distance between the distributions of points in the original high-dimensional space and the reduced low-dimensional space as small as possible. The conditional probability $p(\mathbf{x}_j|\mathbf{x}_i)$ of a point $\mathbf{x}_j$ appearing in the neighborhood of a point $\mathbf{x}_i$ in a high-dimensional space is expressed as

$$p(\mathbf{x}_j|\mathbf{x}_i) = \frac{\exp(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma_i^2})}{\sum_{k \neq i} \exp(-\frac{\|\mathbf{x}_i - \mathbf{x}_k\|^2}{2\sigma_i^2})}, \tag{1}$$

where σ_i is the standard deviation of the Gaussian distribution centered at $\mathbf{x}_i$. It is also symmetrized as the joint probability $p(\mathbf{x}_i, \mathbf{x}_j)$ as

$$p(\mathbf{x}_i, \mathbf{x}_j) = \frac{p(\mathbf{x}_i|\mathbf{x}_j) + p(\mathbf{x}_j|\mathbf{x}_i)}{2n}. \tag{2}$$

On the other hand, the joint probability $q(\mathbf{y}_i, \mathbf{y}_j)$ of points $\mathbf{y}_i$ and $\mathbf{y}_j$ being in a neighborhood in the low-dimensional space is expressed as

$$q(\mathbf{y}_i, \mathbf{y}_j) = \frac{(1 + \|\mathbf{y}_i - \mathbf{y}_j\|^2)^{-1}}{\sum_k \sum_{l \neq k} (1 + \|\mathbf{y}_k - \mathbf{y}_l\|^2)^{-1}}. \tag{3}$$

The optimization is done by minimizing the KL divergence $D_{\mathrm{KL}}(P(\mathcal{X}) \| Q(\mathcal{Y}))$ of probability distributions $P(\mathcal{X})$ and $Q(\mathcal{Y})$ given by probability densities $p(\mathbf{x}_i, \mathbf{x}_j)$ and $q(\mathbf{y}_i, \mathbf{y}_j)$,

$$D_{\mathrm{KL}}(P(\mathcal{X}) \| Q(\mathcal{Y})) = \sum_i \sum_{j \neq i} p(\mathbf{x}_i, \mathbf{x}_j) \log \frac{p(\mathbf{x}_i, \mathbf{x}_j)}{q(\mathbf{y}_i, \mathbf{y}_j)}. \tag{4}$$

The parameter σ_i in Equation 1 is determined by a hyperparameter called perplexity. With perplexity as $Perp$, the parameter σ_i is chosen to satisfy the following equation,

$$Perp = 2^{-\sum_j p(\mathbf{x}_j|\mathbf{x}_i) \log_2 p(\mathbf{x}_j|\mathbf{x}_i)}. \tag{5}$$

The parameter σ_i is smaller in dense areas and larger in sparse areas, but its degree of magnitude is determined by perplexity, so perplexity is interpreted as the number of effective neighbors.

To visualize feature space in the learning process, following ad hoc procedures can be considered utilizing such dimensionality reduction methods. One is performing dimensionality reduction at each time. In this strategy, the positional relationship between objects is maintained at each time, while the position of the same object is not maintained over time. Hence, it is difficult to follow the temporal changes of the objects. Another is applying dimensionality reduction to combined data from all time periods. This strategy is inappropriate because it treats data points originally in different spaces as if they were in the same space.

Dimensionality Reduction of Time Series Data

Dimensionality reduction of time series data can be formulated as follows. At time $t = 1, \ldots, T$, we are given a set $\mathcal{X}[t] = \{\mathbf{x}[t]_i | i = 1, \ldots, n, \mathbf{x}[t]_i \in \mathbf{R}^d\}$ of high-dimensional coordinates corresponding to the set of objects $\mathcal{O}$. Dimensionality reduction of the time series data is the problem of considering a map $f[t] : \mathcal{X}[t] \to \mathcal{Y}[t]$ for $t = 1, \ldots, T$, with $\mathcal{Y}[t] = \{\mathbf{y}[t]_i | i = 1, \ldots, n,\ \mathbf{y}[t]_i \in \mathbb{R}^{d'}, d' < d\}$ being a set of coordinates in the low-dimensional space corresponding to $\mathcal{X}[t]$.

Dynamic t-SNE (Rauber, Falcão, and Telea 2016) is an extension of t-SNE to deal with time series data. Introducing $\mathcal{X}' = \{\mathcal{X}[1], \ldots, \mathcal{X}[T]\}$ and $\mathcal{Y}' = \{\mathcal{Y}[1], \ldots, \mathcal{Y}[T]\}$, the following loss function $L(\mathcal{X}', \mathcal{Y}')$ is minimized by computing the high-dimensional distribution $P(\mathcal{X}[t])$ and the low-dimensional distribution $Q(\mathcal{Y}[t])$ at each time,

$$\begin{aligned} L(\mathcal{X}', \mathcal{Y}') = &\sum_{t=1}^{T} D_{\mathrm{KL}}(P(\mathcal{X}[t]) \| Q(\mathcal{Y}[t])) \\ &+ \frac{\lambda}{2n} \sum_{i=1}^{n} \sum_{t=1}^{T-1} \|\mathbf{y}[t]_i - \mathbf{y}[t+1]_i\|^2. \end{aligned} \tag{6}$$

This is the sum of the KL divergence between the distributions at each time, plus a regularization term that reduces the change in position of each point in the low-dimensional space between consecutive times.

Although it is possible to use dynamic t-SNE to visualize the learning process, but in t-SNE and dynamic t-SNE, the variance structure is not preserved because the range considered as neighborhood changes between sparse and dense regions. Dynamic t-SNE is more restricted in changing the position of points, making it more difficult to capture changes in variance structure compared to t-SNE.

Stochastic Embedding for Multiple Similarities

In this section, we introduce a method for constructing coordinates of time series data from multiple relevances (Sugiura et al. 2022). In addition to the set of objects $\mathcal{O}$, we consider a set of references $\mathcal{U} = \{u_i | i = 1, \ldots, k\}$ such that $o_i \in \mathcal{O}$ is characterized. Suppose that the relevance of o_i and $u_j \in \mathcal{U}$ at time $t = 1, \ldots, T$ is given by a function $\mathrm{rel}(o_i, u_j; t)$. For example, if $\mathcal{O}$ is a set of documents that can be updated over time, such as web pages, and $\mathcal{U}$ is a set of words that characterize the documents, then $\mathrm{rel}(o_i, u_j; t)$ is a function that outputs the number of a word u_j contained in a document o_i at time t. The relevance matrix $R[t] \in \mathbb{R}^{n \times k}$ is used as a matrix summarizing the relevance of each element of $\mathcal{O}$ and $\mathcal{U}$ at time t,

$$(R[t])_{ij} = \mathrm{rel}(o_i, u_j; t). \tag{7}$$

The method aims at obtaining a set of coordinates $\mathcal{Y}[1], \ldots, \mathcal{Y}[T]$ corresponding to the target set $\mathcal{O}$ at time $t = 1, \ldots, T$ and obtaining a set of coordinates $\mathcal{V} = \{\mathbf{v}_i | i = 1, \ldots, k, \mathbf{v}_i \in \mathbb{R}^{d'}\}$ corresponding to the reference set $\mathcal{U}$ for a set of relevance matrices $\mathcal{R} = \{R[1], \ldots, R[T]\}$. In this method, the set of reference coordinates $\mathcal{V}$ is fixed and does not change over time, so changes in relevance over time are reflected in changes in the positional relationship between the time-invariant reference coordinates and the time-variant target coordinates. Here, introduce $Y[t] = (\mathbf{y}[t]_1, \ldots, \mathbf{y}[t]_n)^\top \in \mathbb{R}^{n \times d'}$ as a matrix of elements of the target coordinate set $\mathcal{Y}[t]$ at time t and $V = (\mathbf{v}_1, \ldots, \mathbf{v}_k)^\top \in \mathbb{R}^{k \times d'}$ as a matrix summarizing the elements of the reference coordinate set $\mathcal{V}$. Consider the decomposition of the relevance matrix $R[t]$ as a product of $Y[t]$ and V,

$$R[t] \simeq Y[t] V^\top. \tag{8}$$

For the decomposition, $R[t]$, $Y[t]$ and V are transformed into probability distributions. Here, an exponential family of distributions is used for representing stochasticity. The probability density function $p(o_i, u_j, R[t])$ of the probability distribution $P(\mathcal{O}, \mathcal{U}, R[t])$ corresponding to the relevance $R[t]$

is given by

$$p(o_i, u_j, R[t]) = \exp\{\mathrm{rel}(o_i, u_j; t) - \phi_{o_i}\}, \tag{9}$$

where ϕ_{o_i} is a normalization term for $p(o_i, u_j, R[t])$ to be the conditional probablity density of u_j given o_i. On the other hand, the probability distribution $Q(\mathcal{Y}[t], \mathcal{V})$ corresponding to the pair $Y[t]$ and V of the distribution matrix is defined by a log-bilinear model using the inner product of the target and reference coordinates. The probability density function $q(\mathbf{y}[t]_i, \mathbf{v}_j)$ of the probability distribution $Q(\mathcal{Y}[t], \mathcal{V})$ is given by

$$q(\mathbf{y}[t]_i, \mathbf{v}_j) = \exp(\mathbf{y}[t]_i^\top \mathbf{v}_j - \phi_{\mathbf{y}[t]_i}). \tag{10}$$

Let $L(\mathcal{O},\mathcal{U},\mathcal{R})$ denote the sum of the KL divergence of these probability distributions at time $t = 1, \ldots, T$,

$$L(\mathcal{O},\mathcal{U},\mathcal{R}) = \sum_{t=1}^{T} D_{\mathrm{KL}}(P(\mathcal{O},\mathcal{U}, R[t]) \| Q(\mathcal{Y}[t], \mathcal{V})). \tag{11}$$

The appropriate distribution matrix for the loss function $L(\mathcal{O},\mathcal{U},\mathcal{R})$ is estimated by the optimization as

$$\begin{aligned} &\text{minimize } L(\mathcal{O},\mathcal{U},\mathcal{R}) \\ &\text{subject to } V^\top V = I_{d'}. \end{aligned} \tag{12}$$

These allow us to map multiple relevances into a single space using a log-bilinear model.

The relevance matrix can be seen as a decomposition of the similarity matrix

$$S[t] \simeq R[t]R[t]^\top \in \mathbb{R}^{n\times n}. \tag{13}$$

Here, the similarity $S[t]$ corresponds to the inner product of the coordinates $Y[t]$ via $R[t]$ as

$$\begin{aligned} S[t] &\simeq R[t]R[t]^\top \\ &\simeq Y[t]V^\top V Y[t]^\top \\ &= Y[t]Y[t]^\top. \end{aligned} \tag{14}$$

In visualizing the learning process, we can construct a similarity matrix of the features, and if the similarity matrix can be appropriately decomposed into a relevance matrix, it can be attributed to a mapping of multiple relevances.

Proposed Method

In this study, we construct a similarity matrix of the target features and decompose the similarity matrix into relevance matrices, then attribute it to the mapping of multiple relevances.

At time $t = 1, \ldots, T$, a set of high-dimensional features corresponding to the target set $\mathcal{O}$ is $\mathcal{X}[t] = \{\mathbf{x}[t]_i | i = 1, \ldots, n, \mathbf{x}[t]_i \in \mathbf{R}^d\}$. Consider a map $f[t] : \mathcal{X}[t] \to \mathcal{Y}[t]$ for $t = 1, \ldots, T$, where $\mathcal{Y}[t] = \{\mathbf{y}[t]_i | i = 1, \ldots, n,\ \mathbf{y}[t]_i \in \mathbb{R}^{d'}, d' \leq 3, d' < d\}$ is a set of the coordinates of the low-dimensional space corresponding to $\mathcal{X}[t]$. Suppose that the similarity of $\mathbf{x}[t]_i, \mathbf{x}[t]_j \in \mathcal{X}[t]$ at time $t = 1, \ldots, T$ is given by the function $\mathrm{sim}(\mathbf{x}[t]_i, \mathbf{x}[t]_j)$. For example, the similarity function is given as

$$\mathrm{sim}(\mathbf{x}[t]_i, \mathbf{x}[t]_j) = c\mathbf{x}[t]_i \cdot \mathbf{x}[t]_j, \tag{15}$$

where c is the time-independent constant that normalizes the maximum similarity to 1. Let the similarity matrix $S[t] \in \mathbb{R}^{n\times n}$ be a matrix summarizing the similarity between elements of $\mathcal{X}$ at time t,

$$(S[t])_{ij} = \mathrm{sim}(\mathbf{x}[t]_i, \mathbf{x}[t]_j). \tag{16}$$

Consider decomposing this into a product $R_2[t] \in \mathbb{R}^{n\times n}$ of the relevance matrices $R[t] \in \mathbb{R}^{n\times k}$,

$$S[t] \simeq R[t]R[t]^\top = R_2[t]. \tag{17}$$

In the following, $S[t]$ and $R_2[t]$ are transformed into probability distributions $P(S[t])$ and $Q(R_2[t])$. These probability density functions are given respectively as

$$p(S[t]_{ij}) = \exp\{S[t]_{ij} - \phi_i\}, \tag{18}$$

$$q(R_2[t]_{ij}) = \exp\{R_2[t]_{ij} - \phi_i\}. \tag{19}$$

In the decomposition, we first assume that $R[T]$ has some structure, which we represent by the matrix $B \in \mathbb{R}^{n\times k}$. Next, $R[T]$ and B are transformed into probability distributions $P(B)$ and $Q(R[T])$. These probability density functions are given respectively as

$$p(B_{ij}) = \exp\{B_{ij} - \phi_i\}, \tag{20}$$

$$q(R[t]_{ij}) = \exp\{R[t]_{ij} - \phi_i\}. \tag{21}$$

$R[T]$ is regularized to be similar in structure to B as

$$\begin{aligned} R[T] = \arg\min_{R[T]} &D_{\mathrm{KL}}(P(S[T]) \| Q(R_2[T])) \\ &+ \lambda_1 D_{\mathrm{KL}}(P(B) \| Q(R[T])). \end{aligned} \tag{22}$$

$R[t]$ for $t = 1, \ldots, T-1$ is regularized so that the change from $R[t+1]$ is small as

$$\begin{aligned} R[t] = \arg\min_{R[t]} &D_{\mathrm{KL}}(P(S[t]) \| Q(R_2[t])) \\ &+ \lambda_2 D_{\mathrm{KL}}(Q(R[t+1]) \| Q(R[t])). \end{aligned} \tag{23}$$

The relevance matrix $R[t]$ is decomposed as a product of $Y[t] \in \mathbb{R}^{n\times d'}$ and $V \in \mathbb{R}^{k\times d'}$ in the manner described in the previous section. The learning process in feature space can be visualized as $Y[t]$. If the optimization is done properly, from Equation 14 and 15, the relationship between $X[t]$ and $Y[t]$ is as

$$\begin{aligned} X[t]X[t]^\top &\propto S[t] \\ &\simeq Y[t]Y[t]^\top. \end{aligned} \tag{24}$$

Experiments

In this section, we compare the visualization of artificially generated high-dimensional time series data and high-dimensional features of real data during learning process by the conventional and proposed methods. We compare t-SNE, dynamic t-SNE, and the proposed method. The t-SNE is applied at each time t respectively.

Table 1: Comparison of existing and proposed dimensionality reduction methods

Method	**t-SNE**	**Dynamic t-SNE**	**Proposed method**
Local structure	Preserved	Preserved over time	Preserved over time
Consistent in time direction	No	Yes	Yes
Variance structure	Not preserved	Not preserved	Preserved
Modeling of input space	Distance-based	Distance-based	Similarity-based
Modeling of output space	Student's t-distribution	Student's t-distribution	Log-bilinear model
Main parameters	perplexity	perplexity, λ	λ_1, λ_2, B

Artificial Data: Multivariate Gaussians

As in the evaluation of dynamic t-SNE (Rauber, Falcão, and Telea 2016), we visualize artificial data for the multivariate normal distribution. The data is generated by taking 120 samples from each of three independent normal distributions. Each normal distribution has 10 dimensions, each with a different standard basis vector for the mean and the variance of 0.1 for $t = 1$. Each point in the cluster moves 20% closer to the mean of each cluster with each step from $t = 1$ to $t = 10$.

The perplexity of both t-SNE and dynamic t-SNE is set to 30. The parameter λ of dynamic t-SNE is set to 0.1. The parameters λ_1 and λ_2 of the proposed method are set to 0.1 and 0.3, respectively. Equation 15 is used to calculate the similarity of the proposed method. Let $z_i \in \{0, 1, 2\}$ be the clusters to which the target o_i belongs and $k = 3$ be the number of references, and define the matrix B of the proposed method as

$$\begin{cases} B_{ij} = 1 & (z_i = j - 1) \\ B_{ij} = 0 & (otherwise.) \end{cases}. \tag{25}$$

This B structure is based on the assumption that each cluster is independent at the last time.

The visualization results are shown in Figure 1. Each column corresponds to a) t-SNE, b) dynamic t-SNE, and c) the proposed method, and the rows correspond to $t = 1, 5, 10$ from the top. As shown in the figure, the points in all the methods gradually become closer to the mean of each cluster over time. With t-SNE, the position of each cluster moves with time, whereas dynamic t-SNE and the proposed method show consistency in the position of the clusters. These consistency in the time direction are due to the regularization terms, and the degree to value the consistency depends on the parameter λ of dynamic t-SNE and the parameter λ_2 of the proposed method. In the proposed method, the clusters at $t = 10$ are clearly separated from each other. This may reflect the assumption that the clusters are independent in the structure of B. In such cases where assumptions can be made on the final cluster structure, the proposed method is suitable, and the strength of the constraints can be controlled by λ_1.

We quantitatively compared the time variation of cluster spreading for the dimensionality-reduced data and the original data. Taking the variance-covariance matrix of a cluster at time t as $\Sigma[t] \in \mathbb{R}^{d' \times d'}$, the metrics for the spread of a cluster were calculated as

$$s[t] = \sqrt{\mathrm{tr}\Sigma[t]}. \tag{26}$$

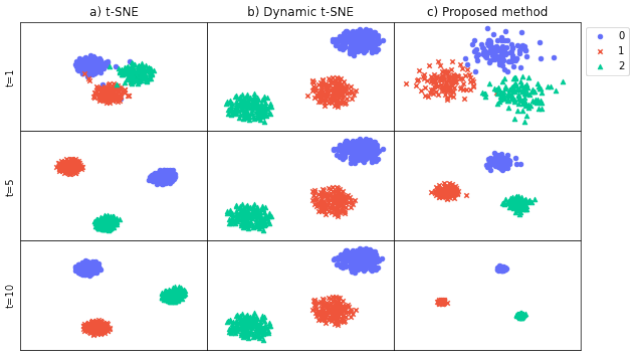

Figure 1: The visualization of multivariate Gaussians

Figure 2 shows, for each method and the original data, the metrics $s[t]$ calculated for all clusters and averaged at each time. In order to compare the reduction ratio for each method, the graph is normalized so that $s[1]$ is 1 for each method and the data before dimensionality reduction. The graph of t-SNE follows the graph of the input space for t less than 4, but does not change much after that. In t-SNE, if the original variance is small for given perplexity, then the probabilities of a point appearing in the neighborhood of another point in a cluster can be all almost the same. In other words, all points in a cluster are treated as being equally nearby. This means that there is effectively a lower bound on the variance that can be captured under a certain perplexity in t-SNE. In this experiment, the variance was small enough when t was greater than 3 under the perplexity of 30, so the variance of neighborhood probabilities within clusters did not change. The graph shows that the metrics $s[t]$ in dynamic t-SNE changes less than the others. Since dynamic t-SNE also has perplexity as a parameter, the variance of clusters that are denser than a certain level will have approximately the same variance after dimensionality reduction. The distortion of the variance structure is more pronounced in dynamic t-SNE compared to t-SNE because it is regularized so that points are less likely to move in the time direction. The graph shows that the clusters in the proposed method shrink at an attenuation rate relatively close to that of the original data. This indicates that the proposed method reduces dimensionality while preserving the distributed structure of the input space. The proposed method is less likely to distort the variance structure of high-dimensional space, because it captures the distance structure between points in

high-dimensional space as a similarity matrix and reflects it in the inner product between points in low-dimensional space, as shown in Equation 14. The temporal consistency regularization does not directly restrict the movement of data points, but rather the change in relevance. Because of the looser restriction compared to dynamic t-SNE, the graph of the proposed method relatively follows the graph of the original data in Figure 2.

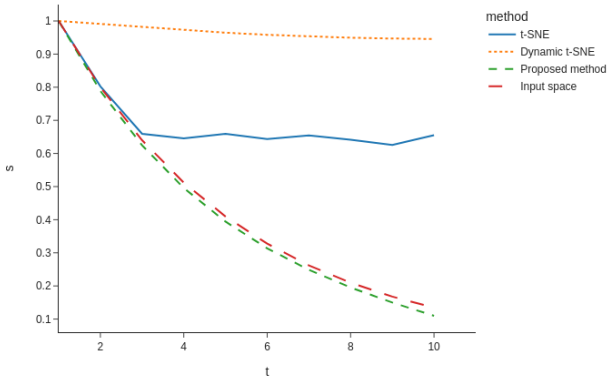

Figure 2: The cluster spreading of multivariate Gaussians

Real Learning Process: MNIST

We visualize the MNIST (Deng 2012) dataset as real data. MNIST is a dataset for image classification of handwritten digits from 0 to 9. Here, we use a total of 500 images, 40 for training and 10 for testing for each class. The model is a 2-layer convolutional neural network (CNN) followed by 2 fully-connected layers, with an output of 10 dimensions, and visualization of the 64-dimensional intermediate representation that serves as the input for the final layer. The training was done for $t = 10$ epochs, and the accuracy of the test data after training was 94%.

In the visualization, the parameters of t-SNE, dynamic t-SNE, and the proposed method are the same as in the previous subsection. Let $z_i \in \{0, 1, \ldots, 9\}$ be the clusters to which the target o_i belongs and $k = 10$ be the number of references, and define the matrix B of the proposed method as in Equation 25.

The visualization results are shown in Figure 3. Each column corresponds to a) t-SNE, b) dynamic t-SNE, and c) the proposed method, and the rows correspond to the intermediate representations after $t = 1, 2, 3, 4$ epochs training from the top. Although all clusters were used in the dimensional reduction, only clusters 0, 1, 3, and 6 are shown in the figure here for ease of viewing. As shown in the figure, all of the visualization methods show the separation of clusters as the learning progresses. The cluster positions are consistent for both dynamic t-SNE and the proposed method, and it is easier to observe the movement of samples between clusters and the separation of clusters during the learning process.

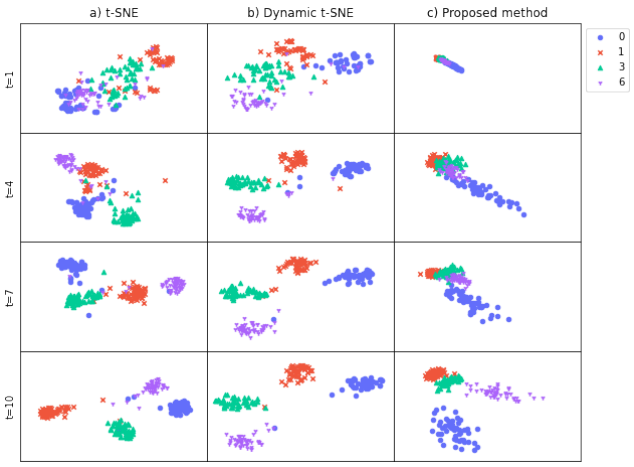

Figure 3: The visualization of MNIST learning process

Figure 4 shows, for each method and the original data, the metrics $s[t]$ of Equation 26 calculated for all clusters and averaged at each time. It can be seen that the variance of each cluster increases during the learning process as indicated by the metrics $s[t]$ of the input space. The graph shows that the metrics $s[t]$ of the proposed method follows the metrics $s[t]$ of the clusters in the input space and grows with time. The visualization using the proposed method in Figure 3 shows that each cluster is separated with increasing variance, which is reasonable considering the trend of variance in the input space in Figure 4. On the other hand, $s[t]$ of t-SNE and dynamic t-SNE do not change much compared to that of the proposed method. In the visualization of these methods in Figure 3, it appears that the variance of each cluster has not changed significantly, which is inconsistent with the metrics $s[t]$ of the original data. From the visualization using the proposed method, it may be possible to suggest improvements, such as speeding up the convergence by setting the initial state to have a larger variance in feature space. On the other hand, existing methods may lead to misinterpretations regarding variance structure.

Conclusion

In this paper, we propose a method of dimensionality reduction for high-dimensional time series data leveraging a log-bilinear model. Similar to dynamic t-SNE, the experiments on both artificial and real data have confirmed that the proposed method is capable of dimensionality reduction while adjusting the trade-off between the consistency in the structure of the original high-dimensional space and that in the time-series direction. We also showed that the proposed method well preserves the distributed structure of the data before dimensionality reduction. The proposed method is theoretically valid in that both spatial structure and temporal direction are constrained by minimizing the distance between probability distributions, and it has an advantage over the existing method in that it can concretely constrain the final cluster structure as a matrix B.

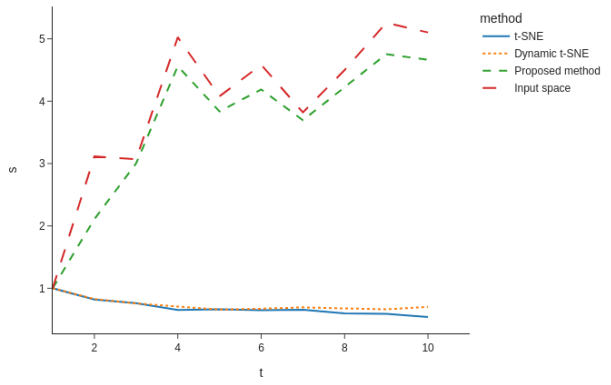

Figure 4: The cluster spreading of MNIST learning process

Future work includes analyzing the learning process of more complex algorithms such as multi-task learning. Since the final cluster structures are not necessarily independent of each other in the target learning algorithms, what assumptions should be appropriate for B is an issue to be addressed in the future.

Acknowledgments

This research was conducted under a contract of "MITIGATE" among "Research and Development for Expansion of Radio Wave Resources(JPJ000254)", which was supported by the Ministry of Internal Affairs and Communications, Japan.

References

Deng, L. 2012. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine* 29(6):141–142.

Rauber, P. E.; Falcão, A. X.; and Telea, A. C. 2016. Visualizing time-dependent data using dynamic t-sne. In *Eurographics Conference on Visualization*.

Sugiura, T.; Okuda, R.; Kodama, A.; Inoue, T.; and Murata, N. 2022. Constructing maps from multiple similarities with stochastic embedding. *The IEICE Transactions (Japanese Edition)* 105(10):525–532.

van der Maaten, L., and Hinton, G. E. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research* 9:2579–2605.

Table 2: List of variables of the proposed method

Variable	**Description**
t	Time
T	Maximum value of t
$\mathcal{O}$	Set of objects to be mapped
n	Number of objects in $\mathcal{O}$
o	Object of $\mathcal{O}$
$\mathcal{U}$	Set of references
k	Number of references
u	Reference of $\mathcal{U}$
$\mathcal{X}$	A set of high-dimensional features corresponding to each of $\mathcal{O}$
d	Dimension of $\mathbf{x}$
$\mathbf{x}$	High-dimensional feature vector of o
$\mathcal{Y}$	A set of low-dimensional coordinates corresponding to each of $\mathcal{O}$
d'	Dimension of $\mathbf{y}$
$\mathbf{y}$	Low-dimensional coordinate vector of o
Y	$(\mathbf{y}_1, \ldots, \mathbf{y}_n)^\top$
$\mathcal{V}$	A set of low-dimensional coordinates corresponding to each of $\mathcal{U}$
$\mathbf{v}$	Low-dimensional coordinate vector of u
V	$(\mathbf{v}_1, \ldots, \mathbf{v}_k)^\top$
$\mathrm{sim}(\mathbf{x}_i, \mathbf{x}_j)$	Similarity of $\mathbf{x}_i and \mathbf{x}_j$
S	A n-by-n matrix of similarity between elements of $\mathcal{X}$
$\mathrm{rel}(o, u; t)$	Relevance of o and u at t
R	A n-by-k matrix of relevance of each element of $\mathcal{O}$ and $\mathcal{U}$
R_2	$RR^\top$, approximation of S
B	A n-by-k matrix to regularize $R[T]$