Serket: A Framework for Construction of Multimodal Learning Models

The University of Electro-Communications
Tomoaki Nakamura
Our group has developed many multimodal learning models based on probabilistic models.
Learning object concepts and language model

- Multimodal information is classified into categories (MLDA)
  - Robot obtains multimodal (visual, auditory, and haptic) information by observing, grasping, and shaking objects
  - User teaches object features by speech

- We assumed robot does not have predefined language knowledge

  Parameters of speech recognition (SR) are learned simultaneously
Spatial Concept Acquisition [Taniguchi+ 2017]

- Taniguchi et al. proposed a more complicated model.
- Robot builds a map using SLAM and simultaneously learns:
  - space name and appearance (MLDA), space region (GMM), parameters of speech recognition (LM)
Multimodal Learning=Complementary Learning

- This model is constructed by connecting two models:
  - Speech recognition and clustering (latent Dirichlet allocation)
  - Shared variable $w^W$ is determined with mutual influence
  - Possibility that speech $o$ is recognized as $w^W$
  - Possibility that $w^W$ is co-occurred with category $Z$

![Diagram of speech recognition and latent Dirichlet allocation models](image)
Background

- These models are also constructed by connecting small-scale models

- Latent variables are shared with two models
- Shared variable is determined with mutual influence
- The parameters are optimized complementarily
Problem in Constructing Models

- These models have complicated structure
- To realize human-like learning models, much more complicated models are required

Formulate the learning system

Drive equations for parameter inference

Implement it with the programing

This part becomes more difficult

- Framework to easily construct models is required
To easily construct multimodal learning models, we have proposed the framework **SERKET**

- Modularizing
- Connecting
- Inference

Shared latent variables are optimized with mutual influence on each model
Modules in SERKET

- Each module has observations and latent variables
- Models are constructed by connecting shared latent variables of modules hierarchically

Parameters are estimated by communication between modules while programmatic independence maintains
Estimation of shared latent variables

- Parameters are estimated by exchanging messages

1. Receive messages from other modules and latent variable $z_{m,n}$ and parameters are updated

2. New messages are sent to other modules based on updated parameters

- By executing this procedure sequentially in each module, the parameters are optimized mutually

- Currently, two methods are implemented for message exchange
Parameter optimization 1

- **Message Passing (MP) Approach**
  - Assume that $z$ is determined with mutual influence:  
    $$ z \sim P(z|\Theta_1, \Theta_2, o) \propto P(z|\Theta_1)P(z|\Theta_2, o) $$
  - **Module 1 → Module 2:**
    - Module 1 sends $P(z|\Theta_1)$ to Module 2
    - Module 2 updates $\Theta_2$ using $P(z|\Theta_1)$
  - **Module 2 → Module 1:**
    - Module 2 sends $P(z|o, \Theta_2)$ to Module 1
    - Module 1 updates $\Theta_1$ using $P(z|o, \Theta_2)$

**Legend:**
- $\Theta_1$ : Parameters of Module 1
- $\Theta_2$ : Parameters of Module 2
- $z$ : Shared latent variables
- $o$ : Observation
Parameter optimization 2

- Sampling Importance Resampling (SIR) approach

- In the case that $w$ has a large number of possibilities such as speech recognition results, Monte Carlo approximation

- Module 2 generates samples $w_n$ and sends them to Module 1
  \[ w_n \sim P(w | \Theta_2, o) \]

- Module 1 resamples based on $P(w | \Theta_1)$ and sends selected samples $w^*$ to Module 2

- Parameters are updated using selected $w^*$
Implementation Examples

- We confirm that integrated models by SERKET improve their performance

- **Used modules:**
  - Variational Auto-Encoder (VAE)
  - Gaussian Mixture Model (GMM)
  - Multimodal Latent Dirichlet Allocation (MLDA)
  - Markov Model (MM)
  - Speech Recognition (SR)

- **Multimodal Dataset:**
  - Image: MNIST
  - Speech: Spoken Arabic Digit Dataset
Implementation Examples

- **Five examples**
  1. VAE+GMM
  2. VAE+GMM+MLDA
  3. VAE+GMM+MLDA+MM
  4. Model for language acquisition by robots
  5. Model for learning object feature extractor by robots

We share Jupyter notebook that you can execute on the Google Collaboratory: [http://iros20.naka-lab.org/](http://iros20.naka-lab.org/)
Example 1: VAE+GMM

- Unsupervised image classification

Image data: MNIST (784 dims)

- Dimensional compression by VAE (18 dims)
- Classification by GMM
Example 1: VAE+GMM

```
import serket as srk
import vae
import gmm
import numpy as np

# Load observations
obs1 = srk.Observation(np.loadtxt("data.txt"))
category = np.loadtxt("category.txt")

# Define modules
vae1 = vae.VAE(18, itr=200, batch_size=500)
gmm1 = gmm.GMM(10, category=category)

# Define connection between modules
vae1.connect(obs1)
gmm1.connect(vae1)

# Optimize the parameters
for i in range(5):
    vae1.update()
    gmm1.update()
```
Example 1: Mutual learning of VAE and GMM

- Modified evidence lower bound (ELBO) of VAE

\[ \mathcal{L}(\theta, \phi; o) = -D_{KL}(q_{\phi}(z_1|o)\|N(\mu, I)) + \mathbb{E}_{q_{\phi}(z_1|o)}[\log p_{\theta}(o|z_1)] \]

Standard VAE: \( \mu = 0 \)

Latent space suitable for classification can be learned with mutual influence

Compute \( \mu \) of cluster

784 dimensional observations are compressed into 18 dimensional \( z_1 \)

\( z_1 \) is estimated using modified ELBO and received \( \mu \)
Example 1: Classification Results

Adjusted Rand Index (ARI)

<table>
<thead>
<tr>
<th>Mutual learning</th>
<th>average</th>
<th>best</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>0.477</td>
<td>0.478</td>
</tr>
<tr>
<td>✓</td>
<td>0.503</td>
<td>0.568</td>
</tr>
</tbody>
</table>

※ average: Average of 10 trials  
best: The best result in 10 trials

Latent variables compressed by PCA

- 18 dims → 2 dims

Confusion Matrix

- Latent space suitable for classification was learned with mutual influence
- Classification accuracy increased
Example 2: VAE+GMM+MLDA

Unsupervised classification of image and speech

Pairwise dataset of image and speech

- Multimodal data is classified in unsupervised manner
Example 2: VAE+GMM+MLDA

Learn correspondence between image and speech

```
import srket as srk
import vae
import gmm
import mlda
import numpy as np

# Load observations
obs1 = srk.Observation(np.loadtxt("data1.txt"))  # 画像
obs2 = srk.Observation(np.loadtxt("data2.txt"))  # 音声
category = np.loadtxt("category.txt")

# Define modules
vae1 = vae.VAE(18, itr=200, batch_size=500)
gmm1 = gmm.GMM(10, category=category)
mlda1 = mlda.MLDA(10, category=category)

# Define connection between modules
vae1.connect(obs1)
gmm1.connect(vae1)
mlda1.connect(obs2, gmm1)

# Optimize the parameters
for i in range(5):
    vae1.update()
gmm1.update()
mlda1.update()
```
Example 2: Classification Results

Adjusted Rand Index (ARI)

<table>
<thead>
<tr>
<th>Mutual learning</th>
<th>average</th>
<th>best</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>0.604</td>
<td>0.638</td>
</tr>
<tr>
<td>✓</td>
<td>0.637</td>
<td>0.735</td>
</tr>
</tbody>
</table>

Classification accuracy improved by **multimodal learning** using images and speech

Confusion Matrix

- **Only image**
  
  - **Multimodal**

  Average: Average of 10 trials
  Best: The best result in 10 trials
Example 3: VAE+GMM+MLDA+MM

- Category and their sequence rules are learned
- Arranged pairwise data in ascending order
- Unsupervised classification of multimodal data
- Learning their transition rules
Example 3: VAE+GMM+MLDA+MM

Suorce code

Learning transition rules

# Load observations
# Define modules
# Define connection between modules
# Optimize the parameters

```
import srk as srk
import vae
import gmm
import mlld
import mm
import numpy as np

# Load observations
obs1 = srk.Observation(np.loadtxt("data1.txt"))  # 画像
obs2 = srk.Observation(np.loadtxt("data2.txt"))  # 音声
category = np.loadtxt("category.txt")

# Define modules
vael = vae.VAE(18, itr=200, batch_size=500)
gmml = gmm.GMM(10, category=category)
mlld1 = mlld.MLDA(10, category=category)
mml = mm.MarkovModel()

# Define connection between modules
vael.connect(obs1)
gmml.connect(vael)
mlld1.connect(obs2, gmml)
mml.connect(mlld1)

# Optimize the parameters
for i in range(5):
    vael.update()
    gmml.update()
    mlld1.update()
    mml.update()
```
Example 3: Classification Results

Adjusted Rand Index (ARI)

<table>
<thead>
<tr>
<th>Mutual learning</th>
<th>average</th>
<th>best</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>0.575</td>
<td>0.524</td>
</tr>
<tr>
<td>✅</td>
<td>0.834</td>
<td>0.980</td>
</tr>
</tbody>
</table>

Classification accuracy significantly improved by learning transition rules

※ average: Average of 10 trials
best: The best result in 10 trials

Confusion Matrix
Conclusion of Example 1, 2 and 3

Adjusted Rand Index (ARI)

<table>
<thead>
<tr>
<th></th>
<th>VAE+GMM</th>
<th>VAE+GMM+MLDA</th>
<th>VAE+GMM+MLDA+MM</th>
</tr>
</thead>
<tbody>
<tr>
<td>average</td>
<td>0.503</td>
<td>0.637</td>
<td>0.834</td>
</tr>
</tbody>
</table>

Confusion Matrix

Complementary learning is realized by using SERKET
Example 4: Model for language acquisition

- **Unsupervised classification of multimodal dataset**
  - Robot obtain multimodal (visual, auditory and haptic) information by observing, grasping and shaking objects
  - User teaches object features by speech
- **We assumed robot does not have language knowledge**
  - Parameters of speech recognition (SR) are learned simultaneously

Learning parameters of speech recognition

This / is / a / bottle

Concepts, word meanings

- Bottle
- Drink
- Plushie
- Soft

Categorization of multimodal information

Observing, grasping, and shaking

User’s utterances

Speech signals

Visual, audio, haptic information

Mutual learning
Example 4: Model for language acquisition

- Learn not only categories but also language model (LM)
  - Language model and object categories are learned mutually

Speech Recognition (SR)

Latent Dirichlet Allocation (LDA)

This is plastic bottle

Recognized strings

Visual, auditory and haptic information

Speech signal

Bottle

Drink
Example 4: Learning of LM

- SIR is used for learning language model

```
MLDA

\{w^{(1)}, \ldots, w^{(L)}\}

- Learn parameters of MLDA
- Compute \( P(w^{(l)}|w^v, w^a, w^t) \)
  \((l = 1, \ldots, L)\)

- \( P(w^{(l)}|w^v, w^a, w^t) \)
- \( W \) is determined by resampling based on \( P(w^{(l)}|w^v, w^a, w^t) \)
- Update parameters of LM

Samples speech recognition results
```

SR
Example 4: Classification Results

- Classify 50 multimodal data

  ![Images of various items for classification](image1)
  ![Images of various items for classification](image2)
  ![Images of various items for classification](image3)
  ![Images of various items for classification](image4)

- Classification accuracy
  - w/o mutual learning: 80%
  - w/ mutual learning: 94%

- Speech recognition accuracy
  - w/o mutual learning: 64%
  - w/ mutual learning: 74%

Classification and speech recognition accuracies improve by multimodal learning based on SERKET
Example 5: Model for learning object feature extractor by robots

- We used a dataset obtained through the robot observed objects and the human taught object feature by speech
  - # of the objects: 499
  - # of the categories: 81

- Words and images were used in this example

  - Words
    - The speech was recognized by phoneme recognizer
    - Recognized strings were segmented into words by unsupervised word segmentation (NPYLM)

  - Images
Example 5: MLDA+VAE

- The model to learn object categories and image features from words and images
Example 5: Classification Results

- Compare with the model where pre-trained CNN is used for feature extractor
- Classification accuracies:
  - Integrated model of VAE and MLDA: 67.8%
  - Model with pre-trained CNN: 66.8%
- Learned latent space:
  - Suitable latent space for classification were learned
Example 5: Cross modal inference

- Images were generated from words input
  - “Cup noodle”
  - “Plastic bottle”
  - “Snack”
  - “Sponge”
- Images that represent the characteristics of the categories were generated
Conclusion

- We implemented VAE, GMM, LDA and MM modules, and integrated model with them.
- We showed implementation examples and it is easy to construct the integrated models.
- Accuracy improved by mutual influence of modules.
- Moreover, we implemented speech recognition module and its integrated model.

Future work

- Integrate deep neural network using Pixyz.
- Improve the efficiency of parameter inference.