確率モデルの統合によるマルチモーダル学習モデルの構築

Construction of Multimodal Learning Models Based on Integrating Stochastic Models

國安 瞭 *1 Ryo Kuniyasu　中村 友昭 *1 Tomoaki Nakamura　長井 隆行 *2 Takayuki Nagai　谷口 忠大 *3 Tadahiro Taniguchi

*1電気通信大学 The University of Electro-Communications　*2大阪大学 Osaka University　*2立命館大学 Ritsumeikan University

In order to realize human-like intelligence artificially, large-scale models are required for robots to understand the environment using multimodal information obtained by various sensors equipped in robots. However, as the scale of models becomes large and complex, it is difficult to construct such models and to derive and implement the equations for their parameter estimation. To overcome this problem, we proposed a framework Serket that makes it easy to construct large-scale models and estimate their parameters by connecting small fundamental models hierarchically while keeping programmatic independence. In this paper, we construct the integrated models of the modules such as variational autoencoder, Gaussian mixture model, Markov model, and multimodal latent Dirichlet allocation, and then show that it is easy to construct the integrated models and their parameters are optimized by communicating between the modules by using Serket.

1. はじめに

人間のような知能を人工的に実現するためには，ロボットに搭載されている様々なセンサから得られるマルチモーダル情報から，ロボットが環境を理解するためのモデルが必要である．そこで，これまで我々は，ロボットが得たマルチモーダル情報を分類することによってロボットが言語や概念を獲得するモデルを提案してきた [Nakamura 14, Taniguchi 16]．人間のような知能を実現するためには，より複雑で大規模なモデルを構築する必要がある．しかし，そのようなモデルのパラメータ推定の式を導出し実装することは，その規模が大きくなるにつれて困難になると考えられる．そこで，我々は小規模なモデルをモジュール化し，階層的に接続することによって大規模なモデルの構築と，そのパラメータ推定を容易に行うことができるフレームワーク Serket (Symbol Emergence in Robotics tool KIT) を提案した [Nakamura 18]．本稿では，文献 [Nakamura 18, 國安 18] で実装した Variational Autoencoder (VAE)[Kingma 13], Gaussian Mixture Model (GMM), Markov Model (MM), Multimodal Latent Dirichlet Allocation (MLDA) のモジュールを Serket を用いて統合し，Serket の有効性を示す．

関連研究として，認知アーキテクチャや確率的プログラミング言語など，様々なモデルを構築するための手法が提案されている [Laird 08, Anderson 09, Patil 10, Tran 16]．しかし，これらの手法では，我々がこれまで提案してきたモデルや大規模なモデルを実装することは難しい．一方，Serket を用いることで，マルチモーダル情報から学習可能な大規模なモデルを構築することが可能となる．

なお本稿の実験で使用したプログラムは全て GitHub[*1] で公開している．

2. Serket

Serket では，小規模なモデルであるモジュールを接続することで大規模なモデルを構築し，モジュール間の通信によりパラメータの推定を行う．本章では，文献 [Nakamura 18] にて提案した Serket に関して説明する．

2.1 モジュール

図 1 に一般化されたモジュールのグラフィカルモデルを示す．各モジュールは，潜在変数 $z_{m,n}$ から生成される他のモジュールと共有される潜在変数 $z_{m-1,*}$，および観測 $\boldsymbol{o}_{m,n,*}$ を持つ．

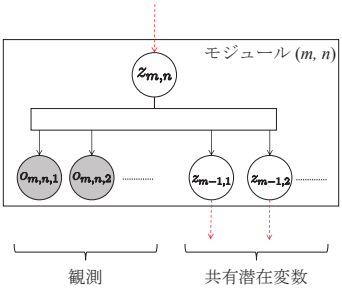


図 1: 一般化されたモジュール

ただし，共有潜在変数や観測を持たないモジュールもこの一般化されたモデルに含まれる．モジュールは以下の要件を満たすことができれば，任意の内部構造を持つことができる．

1. 各モジュールは，次の共有潜在変数が生成される確率などを計算し，潜在変数を共有するモジュールへ送ることができる．

$$P(z_{m-1,i}|z_{m,n}, \boldsymbol{o}_{m,n,1}, \boldsymbol{o}_{m,n,2}, \cdots, \boldsymbol{z}_{m-1}) \quad (1)$$

2. モジュール $(m+1, j)$ から送られる次の確率などを使用して潜在変数 $z_{m,n}$ を決定することができる．

$$P(z_{m,n}|z_{m+1,j}, \boldsymbol{o}_{m+1,j,1}, \boldsymbol{o}_{m+1,j,2}, \cdots, \boldsymbol{z}_{m}) \quad (2)$$

3. 末端のモジュールには共有潜在変数はなく，観測のみを持つ．

このようなモジュールを再帰的に接続することで，大規模なモデルを構築することが可能となる．

2.2 メッセージパッシング法

Serket では，モジュールが相互に影響を及ぼしあいながら共有潜在変数の値が決定される．共有潜在変数を決定する方法はいくつかあり，潜在変数が離散的かつ有限である場合はメッセージパッシング (MP) 法を用いることができる．MP 法を図 2 に示す単純化されたモデルを用いて説明する．モジュール 2 では潜在変数 z_2 から共有潜在変数 z_1 が生成され，モジュール 1 では潜在変数 z_1 から観測 $\boldsymbol{o}$ が生成される．z_1 はモジュール 1 とモジュール 2 で共有されており，この値は次式のように 2 つのモジュールが相互に影響することで決定される．

$$z_1 \sim P(z_1|\boldsymbol{o}, z_2) \quad (3)$$

$$\propto P(z_1|\boldsymbol{o})P(z_1|z_2) \quad (4)$$

連絡先: 國安瞭，電気通信大学，東京都調布市調布ヶ丘 1-5-1，k1512051@edu.cc.uec.ac.jp

*1 https://github.com/naka-lab/Serket

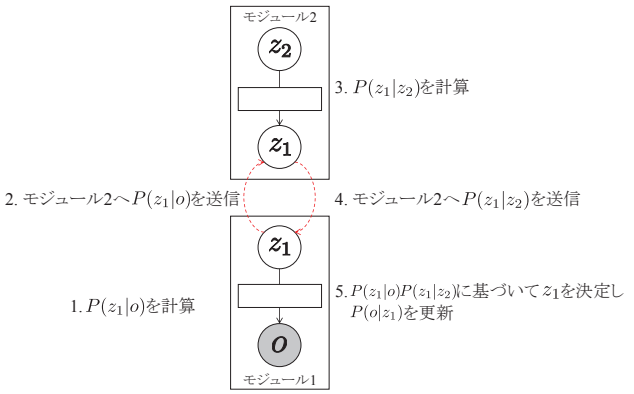


図 2: メッセージパッシング法

$P(z_1|\boldsymbol{o})$ はモジュール 1 で計算され，$P(z_1|z_2)$ はモジュール 2 で計算される．潜在変数が離散的かつ有限の場合，$P(z_1|z_2)$ は z_1 と同じ要素数となる多項分布であり，有限次元のパラメータで表すことができる．したがって，$P(z_1|z_2)$ のパラメータを，モジュール 2 からモジュール 1 へ送ることができる．さらに，モジュール 1 から送られる $P(z_1|\boldsymbol{o})$ を用いてモジュール 2 で $P(z_1|z_2)$ が学習される．これらの分布のパラメータは容易に送受信でき，図 2 の手順で共有潜在変数を決定することができる．このように，メッセージのやりとりによってモデルのパラメータを最適化できるため，本稿ではこの手法を MP 法と呼ぶ．

以降の章では，Serket を用いた実装例を示す．

3. 実装例 1: VAE と GMM の統合モデル

まず，VAE と GMM を統合した次元圧縮と教師なし分類を同時に行うモデルを構築する．

3.1 Serket による実装

図 3 は，VAE と GMM を統合したモデルのグラフィカルモデルである．VAE は，観測 $\boldsymbol{o}$ をエンコーダーにあたるニューラルネットを通して任意の次元の潜在変数 $\boldsymbol{z}_1$ に圧縮し，GMM へ送信する．GMM は，VAE から送られてきた $\boldsymbol{z}_1$ を分類し，分類されたクラスの分布の平均 $\boldsymbol{\mu}$ を VAE へ送信する．Serket では，GMM での分類の影響を受けるため，$\boldsymbol{\mu}$ を用いて VAE の変分下限を以下のように定義する．

$$\mathcal{L}(\boldsymbol{\theta},\boldsymbol{\phi};\boldsymbol{o}) = -wD_{KL}(q_{\boldsymbol{\phi}}(\boldsymbol{z}_1|\boldsymbol{o})\|\mathcal{N}(\boldsymbol{\mu},\boldsymbol{I})) + \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}_1|\boldsymbol{o})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{o}|\boldsymbol{z}_1)] \quad (5)$$

ただし，D_{KL} は KL ダイバージェンスを表しており，w は KL ダイバージェンスの重みである．本稿では，$w = 1$ を用いて実験を行う．これにより，GMM によって同じクラスに分類されたデータの $\boldsymbol{z}_1$ は似た値を持つこととなり，分類に適した潜在空間が学習される．

実際に Serket を用いて実装したソースコードをソースコード 1 に示す．このように，基本的なモジュールを用意することで，容易にモデルを構築することが可能である．

3.2 実験

VAE と GMM を統合したモデルを用いて，MNIST データセットの分類を行なった．データ数は，3,000 である．VAE の潜在変数の次元数は 18 次元とした．分類結果を図 4 に，分類の定量的評価を表 1 に示す．図 4 は分類の混同行列を表しており，縦軸が正解のクラスのインデックス，横軸が分類されたクラスのインデックスである．最適化の際のメッセージのやりとりは 5 回行い，評価指標としては Adjusted Rand Index (ARI)[Hubert 85] を用いた．分類結果および ARI は，10 回の試行の平均と，そのうち最適化後の ARI が最も高かったものを併記している．さらに，VAE により圧縮された潜在変数を，主成分分析でさらに 2 次元に圧縮しプロットしたグラフを図 5 に示す．図 5(a) は VAE 単体で構成された潜在空間であり，図 5(b) が Serket により GMM との相互作用によって構成された潜在空間である．各点の色は，正解のクラスのインデックスを表している．

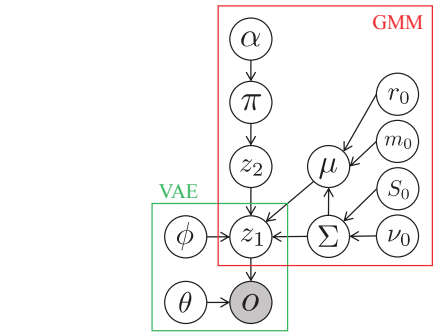


図 3: VAE+GMM のグラフィカルモデル

ソースコード 1: VAE+GMM

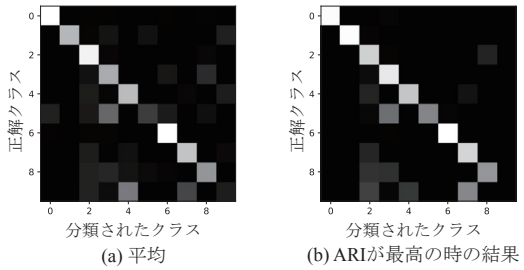
```
import serket as srk
import vae
import gmm
import numpy as np

# 観測と正解ラベルの読み込み
obs1 = srk.Observation(np.loadtxt("data.txt"))
category = np.loadtxt("category.txt")

# モジュールの定義
vae1 = vae.VAE(18, itr=200, batch_size=500)
gmm1 = gmm.GMM(10, category=category)

# モジュールの接続,モデルの構築
vae1.connect(obs1)
gmm1.connect(vae1)

# パラメータの更新,最適化
for it in range(5):
    vae1.update()
    gmm1.update()
```



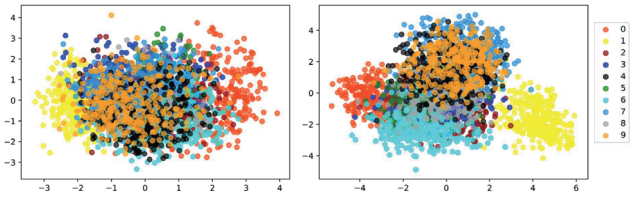
(a) 平均　(b) ARIが最高の時の結果

図 4: VAE+GMM の分類結果 (a)10 回の試行の平均 (b) 最適化後の ARI が最も高かった時の結果

図 4 から 0,1,2,6 は誤分類が少なく高い精度で分類されているが，4,7,9 などは混同し分類精度が低いことが分かる．これは，図 5 からも確認でき，似た特徴を持つデータは潜在空間において近い位置にあるため，同じクラスに誤分類されてしまい Serket による最適化後も近い位置にまとまってしまっている．そのため，ARI の平均値では最適化による大きな向上が見られなかったと考えられる．一方で，図 5 から最適化前では同じクラスであるデータ点が空間上に広く分布しているのに対して，Serket による最適化後ではクラスごとにまとまっている．すなわち，最適化により同じクラスに分類されるデータの潜在変数は似た値を持ち，GMM の分類に適した潜在空間が学習されていることが確認できる．

表 1: VAE+GMM の ARI

	平均値	最高値
最適化前	0.477	0.478
最適化後	0.503	0.568



(a) 最適化前　(b) 最適化後

図 5: 圧縮後の潜在変数

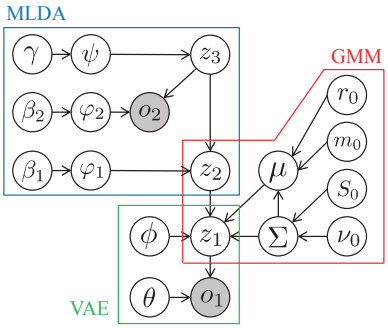


図 6: VAE+GMM+MLDA のグラフィカルモデル

4. 実装例 2: VAE, GMM, MLDA の統合モデル

次に，画像だけでなく音声も用いたマルチモーダル学習モデルを構築する．

4.1 Serket による実装

図 6 は，VAE, GMM, MLDA を統合したモデルのグラフィカルモデルである．GMM は，VAE から送られてきた潜在変数 $\boldsymbol{z}_1$ を分類し，t 番目のデータがクラス $z_{2,t}$ に分類される確率 $P(z_{2,t}|\boldsymbol{z}_{1,t})$ を MLDA へ送信する．MLDA は，GMM から送られてきた確率を用いることで z_2 を観測として扱い，z_2 と観測 $\boldsymbol{o}_2$ を分類し，GMM へ確率 $P(z_{2,t}|z_{3,t}, \boldsymbol{o}_{2,t})$ を送信する．GMM では，送られてきた確率も用いて再度分類を行うことで，MLDA の影響を受け $z_3, \boldsymbol{o}_2$ を考慮した分類が行われる．

ソースコード 2 が実際に実装したソースコードである．このように，モジュールの定義，モジュール間の関係，パラメータの更新の記述を追加することで容易にモデルを拡張することが可能である．

4.2 実験

VAE, GMM, MLDA を統合したモデルを用いて，MNIST データセットおよび Spoken Arabic Digit Data Set の分類を行った．データ数は 3,000 である．Spoken Arabic Digit Data Set は数字の音声発話から MFCC 特徴量を抽出したものであり，本稿では MFCC 特徴量を HAC 特徴量 [Van_hamme 08] に変換して使用した．分類結果を図 7 に，分類の ARI を表 2 に示す．

図 7(a) から 3 章と同様に 4,7,9 が混同しており誤分類が多いが，9 の分類に関して 3 章で構築したモデルより正確に行われており，その他の数字に関しても誤分類が減少していることが確認できる．マルチモーダル情報を用いることによって誤分類が減少し，ARI が向上した．

ソースコード 2: VAE+GMM+MLDA

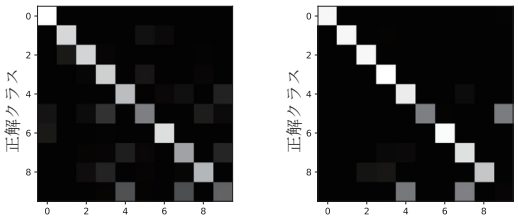
```
import serket as srk
import vae
import gmm
import mlda
import numpy as np

# 観測と正解ラベルの読み込み
obs1=srk.Observation(np.loadtxt("data1.txt")) # 画像
obs2=srk.Observation(np.loadtxt("data2.txt")) # 音声
category = np.loadtxt("category.txt")

# モジュールの定義
vae1 = vae.VAE(18, itr=200, batch_size=500)
gmm1 = gmm.GMM(10, category=category)
mlda1 = mlda.MLDA(10, category=category)

# モジュールの接続,モデルの構築
vae1.connect(obs1)
gmm1.connect(vae1)
mlda1.connect(obs2, gmm1)

# パラメータの更新,最適化
for it in range(5):
    vae1.update()
    gmm1.update()
    mlda1.update()
```



(a) 平均　(b) ARIが最高の時の結果

図 7: VAE+GMM+MLDA の分類結果 (a)10 回の試行の平均 (b) 最適化後の ARI が最も高かった時の結果

表 2: VAE+GMM+MLDA の ARI

	平均値	最高値
最適化前	0.604	0.638
最適化後	0.637	0.735

5. 実装例 3: VAE, GMM, MLDA, MM の統合モデル

さらに，4 章で構築したモデルに MM モジュールを接続することで，時系列データの遷移も学習可能なモデルを構築する．

5.1 Serket による実装

図 8 は，VAE, GMM, MLDA, MM を統合したモデルのグラフィカルモデルである．MLDA は，t 番目のデータがクラス $z_{3,t}$ に分類される確率 $P(z_{3,t}|z_{2,t}, \boldsymbol{o}_{2,t})$ を MM へ送信する．MM は，送られてきた確率 $P(z_{3,t}|z_{2,t}, \boldsymbol{o}_{2,t})$ を用いて繰り返しサンプリングを行い，次のように遷移回数をカウントする．

$$z'_3 \sim P(z_{3,t}|z_{2,t}, \boldsymbol{o}_{2,t}) \tag{6}$$

$$z_3 \sim P(z_{3,t+1}|z_{2,t+1}, \boldsymbol{o}_{2,t+1}) \tag{7}$$

$$N_{z'_3, z_3} + + \tag{8}$$

この値から遷移確率 $P(z_3|z'_3)$ は次のように計算することができる．

$$P(z_3|z'_3) = \frac{N_{z'_3,z_3} + \lambda}{\sum_{\bar{z}_3} N_{z'_3,\bar{z}_3} + K\lambda} \tag{9}$$

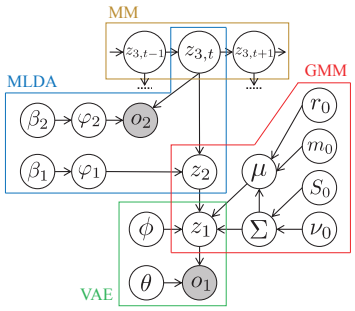


図 8: VAE+GMM+MLDA+MM のグラフィカルモデル

ソースコード 3: VAE+GMM+MLDA+MM

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

# 観測と正解ラベルの読み込み
obs1=srk.Observation(np.loadtxt("data1.txt")) # 画像
obs2=srk.Observation(np.loadtxt("data2.txt")) # 音声
category = np.loadtxt("category.txt")

# モジュールの定義
vae1 = vae.VAE(18, itr=200, batch_size=500)
gmm1 = gmm.GMM(10, category=category)
mlda1 = mlda.MLDA(10, category=category)
mm1 = mm.MarkovModel()

# モジュールの接続,モデルの構築
vae1.connect(obs1)
gmm1.connect(vae1)
mlda1.connect(obs2,gmm1)
mm1.connect(mlda1)

# パラメータの更新,最適化
for it in range(5):
    vae1.update()
    gmm1.update()
    mlda1.update()
    mm1.update()
```

ただし，K は MLDA のクラス数である．この確率を用いて遷移を考慮したそれぞれのクラスに分類される確率を計算し，MLDA へ送信する．MLDA では，送られた確率も用いて再度分類を行うことで，データの遷移を考慮した分類が行われる．ソースコード 3 が実際に実装したソースコードである．

5.2 実験

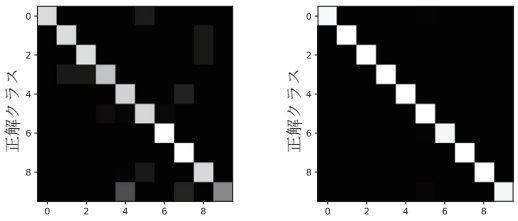
VAE, GMM, MLDA, MM を統合したモデルを用いて，MNIST データセットおよび Spoken Arabic Digit Data Set の分類を行った．データ数は 3,000 であり，それぞれのデータは 0,1,2,3,4,5,6,7,8,9,0,$\cdots$ のように規則的に並び替えて使用した．分類結果を図 9 に，分類の ARI を表 3 に示す．

4 章のモデルにさらに MM を接続することにより遷移を学習することで 4 章までに構築したモデルと比べて誤分類が大幅に減少し，ARI が大きく向上した．このように，Serket を用いてモジュールを接続することにより，それぞれのモジュールが影響を及ぼしあい，モデル全体のパラメータが最適化されていることが確認できる．

6. むすび

本稿では，Serket を用いて VAE, GMM. MM, MLDA を統合したモデルを構築し，Serket の有効性を示した．Serket を用いることでモジュールの定義，モジュール間の関係，パラメータの更新を記述することによって容易にモデルを構築，拡張することが可能である．MP 法によりそれぞれのモジュールが影響を及ぼしあい，モデル全体のパラメータが最適化されることが確認できた．

実験では疑似データを用いたが，本来 Serket はロボットの様々なセンサから得られるマルチモーダル情報から学習するモデルを構築することが目的である．今後，実際にロボットのセンサから得られたマルチモーダル情報を用いた学習モデルの構築と評価や，モジュールの拡充を行っていく予定である．



(a) 平均　(b) ARIが最高の時の結果

図 9: VAE+GMM+MLDA+MM の分類結果 (a)10 回の試行の平均 (b) 最適化後の ARI が最も高かった時の結果

表 3: VAE+GMM+MLDA+MM の ARI

	平均値	最高値
最適化前	0.575	0.524
最適化後	0.834	0.980

謝辞

本研究は JST CREST JPMJCR15E3，JSPS 科研費 JP17K12758 の助成を受け実施したものである．

参考文献

[Anderson 09] Anderson, J. R.: *How can the human mind occur in the physical universe?*, Oxford University Press (2009)

[Hubert 85] Hubert, L. and Arabie, P.: Comparing partitions, *Journal of classification*, Vol. 2, No. 1, pp. 193–218 (1985)

[Kingma 13] Kingma, D. P. and Welling, M.: Auto-encoding variational bayes, *arXiv preprint arXiv:1312.6114*, pp. 1–14 (2013)

[Laird 08] Laird, J. E.: Extending the Soar cognitive architecture, *Frontiers in Artificial Intelligence and Applications*, Vol. 171, p. 224 (2008)

[Nakamura 14] Nakamura, T., Nagai, T., Funakoshi, K., Nagasaka, S., Taniguchi, T., and Iwahashi, N.: Mutual learning of an object concept and language model based on MLDA and NPYLM, in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 600–607 (2014)

[Nakamura 18] Nakamura, T., Nagai, T., and Taniguchi, T.: SERKET: An Architecture for Connecting Stochastic Models to Realize a Large-Scale Cognitive Model, *Frontiers in Neurorobotics*, Vol. 12, pp. 1–16 (2018)

[Patil 10] Patil, A., Huard, D., and Fonnesbeck, C. J.: PyMC: Bayesian stochastic modelling in Python, *Journal of statistical software*, Vol. 35, No. 4, p. 1 (2010)

[Taniguchi 16] Taniguchi, A., Taniguchi, T., and Inamura, T.: Spatial concept acquisition for a mobile robot that integrates self-localization and unsupervised word discovery from spoken sentences, *IEEE Transactions on Cognitive and Developmental Systems*, Vol. 8, No. 4, pp. 285–297 (2016)

[Tran 16] Tran, D., Kucukelbir, A., Dieng, A. B., Rudolph, M., Liang, D., and Blei, D. M.: Edward: A library for probabilistic modeling, inference, and criticism, *arXiv preprint arXiv:1610.09787*, pp. 1–33 (2016)

[Van_hamme 08] Van_hamme, H.: Hac-models: A novel approach to continuous speech recognition, in *Ninth Annual Conference of the International Speech Communication Association* (2008)

[國安 18] 國安 瞭, 中村 友昭, 青木 達哉, 谷口 彰, 尾崎 僚, 伊志嶺 朝良, 横山 裕樹, 小椋 忠志, 長井 隆行, 谷口 忠大：確率モデルの統合による大規模なモデルの実現–VAE, GMM, HMM, MLDA の統合モデルの実装と評価–, 情報論的学習理論ワークショップ, T-34 (2018)