공학박사학위논문

스트림 데이터에서 시공간적 패턴 학습을 위한 비모수적 확률 모델

Nonparametric Probabilistic Models for Learning Spatiotemporal Patterns of Stream Data

2012년 8월

서울대학교 대학원
컴퓨터공학부

석 호 식

스트림 데이터에서 시공간적 패턴 학습을 위한 비모수적 확률 모델

(Nonparametric Probabilistic Models for Learning Spatiotemporal Patterns of Stream Data)

指導敎授 張 炳 卓

이 論文을 工學博士 學位論文으로 提出함

2012년 4월

서울대학교 대학원

컴퓨터공학부

石 皓 植

石皓植의 工學博士 學位論文을 認准함

2012년 5월

委 員 長 _____

副委員長 _____

委 員 ROBERT IAN MCKAY _____

委 員 KYU-BAEK HWANG _____

委 員 _____

Abstract

The complexity in temporal domains requires introduction of the unobservable inherent dependencies. However, this task is very challenging due to difficulties such as hierarchical characteristics of the domain and abrupt changes in modalities. This thesis addresses the problem of extracting inherent dependencies in temporal domains. We develop algorithms for modeling and inference: inference with changing underlying distributions, temporal sequence learning with summarized problem space, and inference in a multichannel sequential environment.

There exist domains where it is not possible to select a stable underlying distribution due to environmental causes. We develop an inference algorithm based on the feature relevance network in order to tackle the distribution change. The idea of feature relevance network is inspired by the existence of stable relations between features irrespective of environmental change. The proposed algorithm is verified on an indoor location estimation task. We show that the non-parametric approach making no assumption on feature distributions is capable of discovering hidden relations among features.

The most obvious instances of temporal domain are dynamic streams such as TV drama. In order to estimate hidden dependencies in an episode of TV drama and represent a whole episode in a more succinct form, this thesis introduces a temporal learning scheme based on a set of particles. Instead of assuming a prior distribution, particles capture prominent characteristics of a given stream in a collaborative manner. The proposed method evolves particles through two-stage learning. At the first stage, a segment (scene) is estimated using evolutionary particle filtering (PF). At the second stage, a transitional probability matrix representing dependencies between estimated segments is computed and stored. We demonstrate performance by

comparing to human-evaluated ground truth and regenerating images of expected sequences succeeding a given seed image.

For inference in a multichannel sequential environment, the sequential hierarchical Dirichlet process (sHDP) is developed. For seamless processing of multichannel data, sHDP divides a multichannel stream into sub-channel of single modality and builds a latent model for a sub-channel. Changes in sub-channels are linked though a dynamic channel merging scheme. The proposed method is applied for semantic segmentation in real TV drama episodes and the performance is compared to human evaluated ground-truth.

By incorporating inherent dependencies, we present successful algorithms to deal with applications in temporal domains. The three areas of this thesis are promising realizations of inherent dependencies learning, and the algorithms presented here inform the range of possibility of applicability in temporal domains.

Keywords: **Probabilistic Model, Nonparametric, Spatiotemporal Patterns, Evolutionary Particle Filtering, Stream Analysis**

Student Number: 2004-31034

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Background and Motivation

Most of phenomena in real world are temporal ones. If we knew the forms of the underlying density functions of phenomena, it is possible to treat these phenomena using some well-established methods for estimating the *parameters* of a underlying density function. However, in most real world problems this assumption is suspicious; the common parametric forms rarely fit the densities actually encountered in practice (Duda et al., 2001). In order to deal with non-linear temporal processes, classical methods rely on defining a fixed, finite set of models with known parameters: utilizing knowledge of the number of mixtures and estimating the model parameters from data, or assuming deterministic dynamics.

These approaches raise three questions; how to utilize unchanging properties in dynamic phenomena, is it possible to analyze temporal data without relying on unrealistic assumptions, and is there any way to get more flexible models? The first question is an attempt to deal with domains where it is very unrealistic to assume underlying distributions but it is possible to observe some unchanging characteris-

tics. The second question is about how to analyze data without rigid prior knowledge on a domain. The third question relates to situations where the assumption of any fixed number of mixtures is unrealistic.

The situations where underlying distributions are changing with time is a typical domain of transfer learning (Caruana, 1997). However, one cannot expect reliable estimation when changes in underlying distributions are unpredictable. In this thesis, we introduce an inference method in a changing world focusing on inter-feature relationships resilient to distributional changes among features. A dynamic domain under influence of non-linear dynamic phenomena could be handled by particle filters (Arulampalam et al., 2002; Pitt, 2002). As a motivating example, temporal domains of TV drama are tackled. Based on the particle filters, we introduce a model to learn inherent dependency structures in a temporal stream. A temporal stream such as a TV drama is composed of multiple modality channels and there exist hierarchical structures in these sub-channels. For these domains, the clustering properties induced by the Dirichlet process prior have been exploited in many applications. Especially, a hierarchical extension of Dirichlet process such as the hierarchical Dirichlet process (Teh et al., 2005) has proven its usefulness in a variety applications (Ren et al., 2008; Fox et al., 2008; Orbanz et al., 2007). As a final contribution of this thesis, we discuss a scheme for approximating semantic changes in a TV drama based on properties of HDP.

1.2 Problems to be Tackled

In this thesis, three problems will be tackled. The first problem is about a question of how to encounter unfixed underlying density functions with some features remain unchanged. The second problem raises a question of how to analyze dynamic temporal patterns without employing sophisticated prior assumptions. Finally, the

third problem attempts to answer a question opposite to the second question: how to detect unknown number of changes with models allowing the number of latent variables to grow as necessary, but where individual variables still follow some prior distributions.

The task of indoor location estimation based on Wi-Fi signal provides useful data set for an environment with changing underlying distributions (Yang et al., 2008). Wi-Fi data is noisy owing to the indoor environment's multipath and shadow fading effects. The data distribution changes constantly as people move and as temperature and humidity change. This is typical example of a non-parametric approach is required.

The task of analyzing dynamic temporal patterns presents insight for analyzing real-world temporal streams such TV dramas. In TV dramas, the number of changes are unknown and a change occurs very abruptly. In addition, it is very unrealistic to assume that changes in a TV drama follow a distribution. Although various analyzing methods such as DTW (Dynamic Time Warping) (Kruskal and Liberman, 1999) exist, the segmentation and comparison of multimodal streams remain as challenging tasks.

1.3 Our Approach and Its Contributions

Our contributions in this dissertation is as following. Firstly, we propose a new learning method based on feature relevance network. Secondly, we propose an extension of evolutionary particle filter (Kwok et al., 2005; Park et al., 2009) for analyzing temporal dependencies in a temporal stream. Finally, we propose a sequential HDP (sHDP) scheme for semantic segmentation of a temporal domain.

A feature relevance network is for inferencing in a situation with unpredictably changing underlying distributions. The proposed method focuses on adjacent fea-

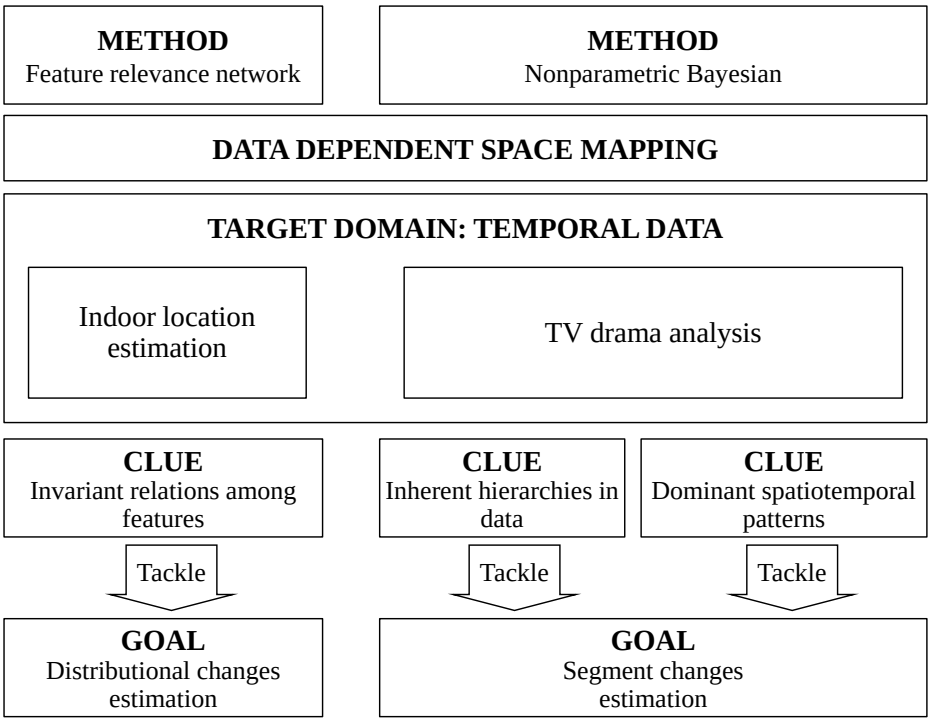


Figure 1.1: Our approach for nonparametric analysis of temporal data. In order to estimate indoor location, we need to cope with distributional changes due to environmental factors. For TV drama analysis, we need to detect segment changes. We tackle these challenges by focusing on following characteristics: invariant relations among features, inherent hierarchies in data, and existence of dominant spatiotemporal patterns.

ture pairs. These adjacent feature pairs constitute edges of a graph structure and then, a problem space is reconstructed based on this graph structure. When a new instance is given, it is mapped onto the new problem space and the cost of map-

ping are computed. Based on the proposed feature relevance network, we show that the proposed method could deal with a problem where datasets are obtained from different distributions.

Evolutionary particle filters are promising methods for dynamic domains. We introduce a method capable of segmenting a given stream and learning inherent temporal dependencies. In the case of multi-modal stream data, it is difficult to derive a single data type efficiently representing various data structures in a multi-modal stream. However, a group of particles has the potential to represent a whole stream more adaptively than a single data type although each particle has very limited representation power if these particles collaborate to represent a frame. The proposed method extracts hidden dependencies in a multi-modal stream while maintaining the diversity in a particle population by altering structure and introducing new particles through genetic operations. We apply the proposed method on segmentation of dominant images in a TV drama.

Temporal and multi-modal streams such as a TV drama has various characteristics. There is temporal consecutiveness one could exploit and conceptual hierarchies composed of unlimited number of image features, words, objects, sentences, and stories (Mittal, 2006). We propose a method for approximating sematic changes in a TV drama focusing on hierarchies in multi-channels. For analyzing an image channel, the proposed method utilizes hierarchical structures inherent in a domain. In order to deal with a sound channel, a dialogue interval is estimated based on MFCC (Mel Frequency Cepstral Coefficient). We estimates changes in an image channel by constructing latent models for each segment. Changes in a sound channel is estimated by utilizing mute intervals in a dialogue. By merging estiated changes in each channel dynamically, we are able to make semantic segments in episodes of a TV drama.

1.4 Thesis Organization

This dissertation is organized as follows.

In Chapter 2, we discuss probabilistic models in dynamic domains. First, we describe dynamic inference methods, particle filters and its extension: evolutionary particle filtering. Then, Bayesian frameworks for dynamic environments is explained. Finally, we describe a general framework for dynamic domains.

In Chapter 3, we propose a feature relevance network model for a domain where underlying distributions are changing unpredictably. First, we define the proposed feature relevance network framework. The unique points of feature relevance network model are unchanging inter-relatedness and non-parametric approaches making no assumptions about signal distributions. Then, a learning algorithm for feature relevance network is introduced. As a real world application of feature relevance network, the task of location estimation based on Wi-Fi signal is tackled.

Chapter 4 describes a temporal sequence learning model based on evolutionary particle filters. Contrary to previous researches, we do not employ sophisticated prior assumptions. The proposed method aims to detect changes in dominant images in a stream. First, we propose a population-based representational scheme for detecting changes. In order to alleviate problem of prematurely convergence, we introduce diversity through genetic operation. Second, a learning algorithm for dependencies relationships is introduced. The proposed method is applied to the task of detecting dominant image changes and learning of hidden dependencies a TV drama.

In Chapter 5, we introduce a nonparametric model for approximating semantic changes in a video stream. For analyzing streams, we focus on inherent hierarchical structures in a video stream and multi-channels in a stream. Rather than utilizing a conceptual dictionary, we construct a temporal model explaining changes in a channel. First, we propose a sequential HDP (Hierarchical Dirichlet Process) based

on Dirichlet process for an image channel. The proposed sequential HDP constructs a latent model for each segment and changes are estimated based on the likelihood of a frame given the current latent model. For a sound channel, speech intervals are estimated. The estimated changes in each channel are mixed dynamically. As a real world application, semantic changes in episodes of a TV drama is estimated based on the proposed method.

Finally, we summarize the dissertation and discuss its contributions in Chapter 6.

Chapter 2

Probabilistic Models

In this chapter, we review the related methodologies upon which our contributions are based. We begin in Section 2.1 by introducing particle filtering and the evolutionary particle filtering that allows for the development of efficient inference techniques without assuming sophisticated priors. We then describe a nonparametric Bayesian framework: the Dirichlet process and its hierarchical extension in Section 2.2.

In Section 2.3, we propose a general framework in which the proposed methods in Chapter 3, Chapter 4, and Chapter 5 are combined.

2.1 Particle Filtering

Particle filters generate approximations to filtering distributions and are commonly used in non-linear and/or non-Gaussian state space models. We discuss general concepts associated with particle filtering, provide an overview of the main particle filtering algorithms, and its evolutionary extension.

2.1.1 Particle Filters

In order to analyze and make inference about a dynamic system, at least two models are required. First, a model describing the evolution of the state with time and, second, a model relating the noisy measurements to the state (Arulampalam et al., 2002; Pitt, 2002). Simulation based filters are based on the principle of recursively approximating the filtering density by a large samples with weights. Particle filters are used to estimate Bayesian models in which the latent variables are connected in a Markov chain. Particle filters aims to estimate the sequence of hidden parameters $\mathbf{x}_k$ (a state) based only on the observed data.

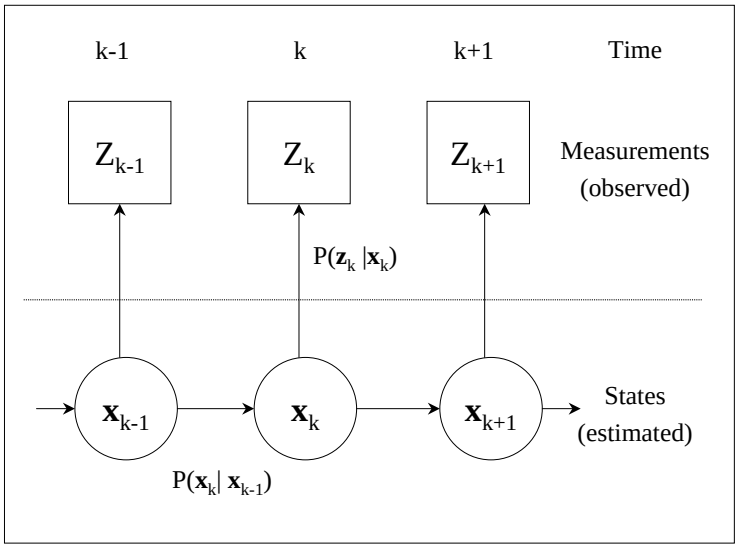


Figure 2.1: Assumptions in particle filters. The observations are conditionally independent given the sate: $p(\mathbf{z}_k|\mathbf{x}_k)$. State transition probability is defined from $p(\mathbf{x}_k|\mathbf{x}_{k-1})$

To define the problem, consider the evolution of the state sequence depicted by

Eq. 2.1:

$$\mathbf{x}_k = f_k(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{v}_{k-1}) \tag{2.1}$$

where $f(\cdot,\cdot,\cdot)$ is an evolution function, $\mathbf{x}_k,\ \mathbf{x}_{k-1} \in \mathbb{R}^n$ are current and previous state, $\mathbf{v}_{k-1}$ means state noise, and $\mathbf{u}_{k-1}$ is a known input.

The observed values are obtained from measurement function:

$$\mathbf{z}_k = h_k(\mathbf{x}_k, \mathbf{u}_k, \mathbf{n}_k) \tag{2.2}$$

where, $h(\cdot,\cdot,\cdot)$ is a measurement function, $\mathbf{z}_k$ means a measurement, and $\mathbf{n}_k$ is a measurement noise.

In order to estimate $\mathbf{x}_k$, we attempt to construct the posterior $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ instead of full posterior $p(\mathbf{x}_{1:k}|\mathbf{z}_{1:k})$. This construction process is composed of prediction step and updated step:

$$\textbf{Prediction step}: \ p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1}) \rightarrow p(\mathbf{x}_k|\mathbf{z}_{1:k-1})$$

$$\textbf{Update step}: \ p(\mathbf{x}_k|\mathbf{z}_{1:k-1}), \mathbf{z}_k \rightarrow p(\mathbf{x}_k|\mathbf{z}_{1:k})$$

For this recursive process, ideal value is obtained from Eq. 2.3 and Eq. 2.4:

$$p(\mathbf{x}_k|\mathbf{z}_{1:k-1}) = \int p(\mathbf{x}_k|\mathbf{x}_{k-1}) p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1}) d\mathbf{x}_{k-1} \tag{2.3}$$

$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) = \frac{p(\mathbf{z}_k|\mathbf{x}_k) p(\mathbf{x}_k|\mathbf{z}_{1:k-1})}{p(\mathbf{z}_k|\mathbf{z}_{1:k-1})} \tag{2.4}$$

The problem is that integrals in the above process is not tractable, so a sequential importance sampling is proposed.

Sequential importance sampling

The sequential importance sampling (SIS) is a Monte Carlo (MC) method that forms the basis for most sequential MC filters developed over the past decades (Arulampalam et al., 2002). In order to develop the details of the algorithm, let $\{\mathbf{x}^i_{0:k}\}$

denote a set of support points (particles) and $\{w_k^i\}$ is a set of associated weights. Then, the posterior density at k can be approximated as

$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) \approx \sum_{i=1}^{N_s} w_k^i \delta(\mathbf{x}_{0:k} - \mathbf{x}_{0:k}^i) \tag{2.5}$$

The weights are chosen using the principle of *importance sampling*. This principle relies on the following. Suppose we sample directly from a (different) *importance function* $q(\cdot)$. Our approximation is still correct if

$$w_k^i \propto \frac{p(\mathbf{x}_{0:k}^i|\mathbf{z}_{1:k})}{q(\mathbf{x}_{0:k}^i|\mathbf{z}_{1:k})} \tag{2.6}$$

If the importance density is chosen to factorize such that

$$q(\mathbf{x}_{0:k}|\mathbf{z}_{1:k}) = q(\mathbf{x}_k|\mathbf{x}_{0:k-1}, \mathbf{z}_{1:k}) q(\mathbf{x}_{0:k-1}|\mathbf{z}_{1:k-1})$$

then one can obtain samples $\mathbf{x}_{0:k}^i \sim q(\mathbf{x}_{0:k}|\mathbf{z}_{1:k})$ by augmenting each of the existing samples $\mathbf{x}_{0:k-1}^i \sim q(\mathbf{x}_{0:k-1}|\mathbf{z}_{1:k-1})$ with the new state $\mathbf{x}_k^i \sim q(\mathbf{x}_k|\mathbf{x}_{0:k-1}, \mathbf{z}_{1:k})$. Then, weight update equation can be shown to be

$$w_k^i = w_{k-1}^i \frac{p(\mathbf{z}_k|\mathbf{x}_k^i) p(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i)}{q(\mathbf{x}_k^i|\mathbf{x}_{0:k-1}^i, \mathbf{z}_{1:k})} \tag{2.7}$$

The problem with SIS approach is that after a few iterations, most particles have negligible weight (the weight is concentrated on a few particles only). This phenomena is known as a degeneracy problem. The effects of degeneracy can be reduced by resampling whenever a significant degeneracy is observed. The basic idea resampling is to replace old set of samples with new set of samples, such that sample density better reflects posterior probability density function.

2.1.2 Evolutionary Particle Filtering

Particle filters are suitable for nonlinear estimation and widely used for dynamic systems (Rekleitis, 2004). However, there are some cases in which most particles

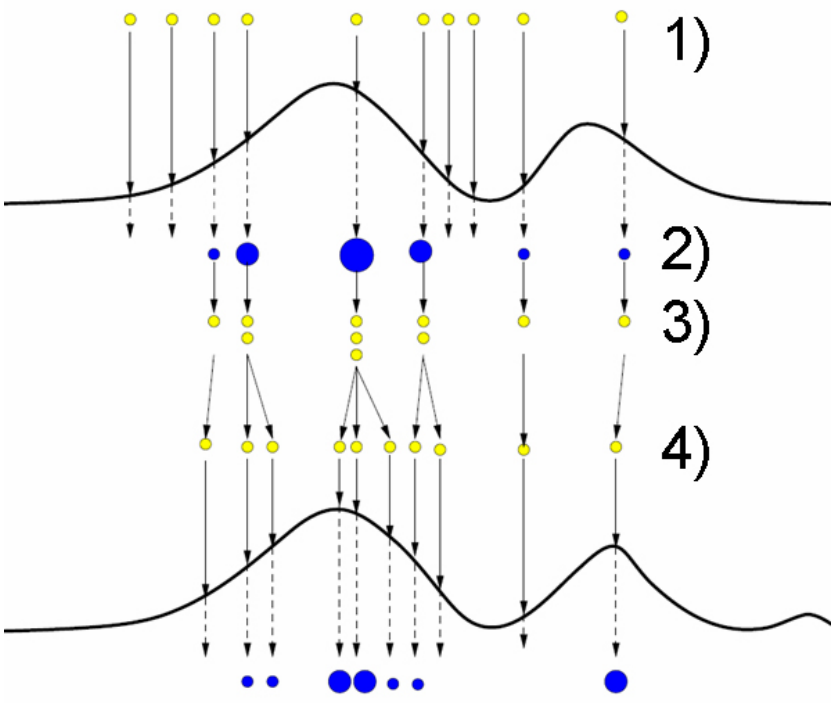


Figure 2.2: The resampling principle. 1) $\{\tilde{\mathbf{x}}_{t-1}^{(i)}, {N_s}^{-1}\}$, 2) $\{\tilde{\mathbf{x}}_{t-1}^{(i)}, \tilde{w}_{t-1}^{i}\}$, 3) $\{\mathbf{x}_{t-1}^{(i)}, N_s^{-1}\}$, $\{\tilde{\mathbf{x}}_{t}^{(i)}, N_s^{(-1)}\}$ (image derived from (van der Merwe et al., 2000)).

are concentrated prematurely at a wrong point (Park et al., 2009). In order to preserve diversity, a modified scheme, evolutionary particle filtering, is introduced. In evolutionary particle filtering, genetic algorithms (GA) are incorporated into a particle filter (Park et al., 2009; Higuchi, 1997; Kwok et al., 2005).

Algorithm 1 explains the overall evolutionary particle filtering (EPF) procedure. In order to overcome the premature concentrations, EPF adopts crossovers and mutation operators. According to a specified probabilities π_c and π_m, new particles are generated. After sampling particles, weights are assigned according to Eq. 2.8.

Algorithm 1 Evolutionary Particle Filter

Sampling the particle $\{\mathbf{x}_k^i\}$ are sampled from one-time ahead particles $\{\mathbf{x}_{k-1}^i\}$

- Sampled according to the state transitional probability $p(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i)$
- New particle pairs are generated from crossovers
- New particle is generated from mutation

Importance Weighting Assigning to each particle a weight

$$w_k^i \propto w_{k-1}^i \frac{p(y_k|\mathbf{x}_k^i)p(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i)}{(1-\pi_c-\pi_m)p(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i)+\pi_c p(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i,\mathbf{x}_{k-1}^j)+\pi_m p(\mathbf{x}_k^i|\mathbf{x}_{k-1}^j)} \quad (2.8)$$

where, π_c and π_m are the crossover and mutation probabilities, respectively.

Resampling

If $N_{eff} < N_{thr}$, normalize importance weight and resample $\mathbf{x}_k^i$ by w_k^i

Return $\{\mathbf{x}_k^i, w_k^i\}_{i=1}^N$

Here, N_{eff} is a diversity measure.

Resampling would be introduced if a diversity value is lower than a threshold.

By introducing genetic operators, it is possible to avoid the premature convergence. EPF based representation is also useful for image processing. If one tries to represent an image using a particle, the resulted representation is likely to be vulnerable to minor distortions. The distributed nature of Genetic Algorithms solves this dilemma by collective representation of particle groups.

2.2 Nonparametric Bayesian Framework

A Bayesian nonparametric model is a Bayesian model on an infinite-dimensional parameter space. The parameter space is typically chosen as the set of all possible

solutions for a given learning problem. A Bayesian nonparametric model uses only a finite subset of the available parameter dimensions to explain a finite sample of observations, with the set of dimensions chosen depending on the sample, such that the effective complexity of the model adapts to the data (Orbanz and Teh, 2010). It is desirable to consider models that are not limited to finite parameterizations. Bayesian nonparametric methods avoid the often restrictive assumptions of parametric models by defining distributions on function space such as that of probability measures. If suitably designed, these method allow for efficient, data-driven posterior inference (Fox, 2008). In the following sections, we briefly describe some classes of Bayesian nonparametric methods: the Dirichlet process and its hierarchical extension.

2.2.1 Dirichlet Process

A Dirichlet process is a stochastic process that defines a distribution on discrete measures. It is possible to describe a discrete measure G in terms of a set of atoms δ_{θ_k} weights β_k, with

$$G = \sum_k \beta_k \delta_{\theta_k} \tag{2.9}$$

. The Dirichlet process is defined by a *base measure* H on Θ and a *concentration parameter* γ (Canini and Griffiths, 2011). Let (Θ, β) be a measurable space, with G_0 a probability measure on the space. Let α_0 be a positive real number. A *Dirichlet process* $DP(\alpha_0, G_0)$ is defined to be the distribution of a random probability measure G over (Θ, β) such that, for any finite measurable partition $(A_1, ..., A_r)$ of Θ, the random vector$(G(A_1), ..., G(A_r))$ is distributed as a finite-dimensional Dirichlet distribution with parameters $(\alpha_0 G_0(A_1), ..., \alpha_0 G_0(A_r))$:

$$(G(A_1), ..., G(A_r)) \sim Dir(\alpha_0 G_0(A_1), ..., \alpha_0 G_0(A_r)) \tag{2.10}$$

The Dirichlet process can be understood on two different perspectives - one based on the stick-breaking construction and one based on the Chinese restaurant process (Teh et al., 2005).

Stick-breaking Construction

The stick breaking construction is based on independent sequences of independent random variables $(\pi'_k)_{k=1}^{\infty}$ and $(\theta_k)_{k=1}^{\infty}$:

$$\pi'_k|\alpha_0, G_0 \sim Beta(1, \alpha_0) \quad \theta_k|\alpha_0, G_0 \sim G_0, \tag{2.11}$$

where $Beta(a, b)$ is the Beta distribution with parameter a and b. Now define a random measure G as

$$\pi_k = \pi'_k \prod_{l=1}^{k-1}(1 - \pi'_i) \quad G = \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k}, \tag{2.12}$$

where δ_θ is a probability measure concentrated at θ. Sethuraman (1994) showed that G as defined in this way is a random probability measure distributed according to $DP(\alpha_0, G_0)$

Chinese Restaurant Process

Another perspective on the Dirichlet process is based on the Pólya urn scheme by Blackwell and MacQueen (1973). Consider a set of observations $\{\phi_i\}_{i=1}^{N}$ such that $\phi_i|G_0 \sim G_0$. The Pólya urn scheme shows that not only are draws from the Dirichlet process discrete, but also that they exhibit a clustering property. Blackwell and MacQueen (1973) introduced that a Pólya urn representation of the ϕ_i that results from integrating over the underlying random measure G_0:

$$\phi_i|\phi_1, ..., \phi_{i-1}, \alpha_0, G_0 \sim \sum_{l=1}^{i-1} \frac{1}{i - 1 + \alpha_0} \delta_{\theta_i} + \frac{\alpha_0}{i - 1 + \alpha_0} G_0 \tag{2.13}$$

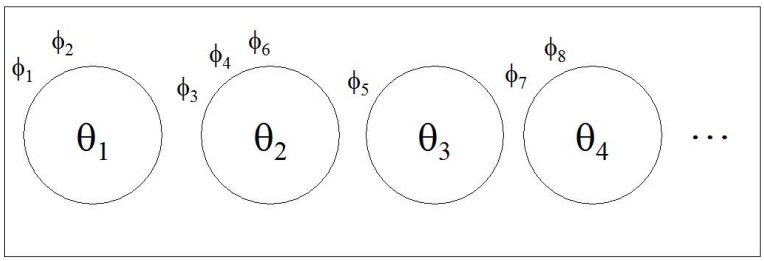


Figure 2.3: A depiction of a Chinese restaurant after eight customers have been seated. Customers (ϕ_i's) are seated at tables (circles) which correspond to the unique values θ_k (image derived from (Teh et al., 2005)).

This expression shows that ϕ_i has positive probability of being equal to one of the previous draws, and there is a positive reinforcement effect - the more often a point is drawn, the more likely it is drawn in the future.

With a new set of variables that represent distinct values of the atoms, it is possible to make the clustering property explicit. Define $\theta_1, ..., \theta_K$ to be the distinct values taken on by $\phi_1, ..., \phi_{i-1}$, and let n_k be the number of values $\phi_{i'}$ that are equal to θ_k for $1 \leq i' < i$. Then Eq. 2.13 can be re-expressed as

$$\phi_i | \phi_1, ..., \phi_{i-1}, \alpha_0, G_0 \sim \sum_{k=1}^{K} \frac{n_k}{i - 1 + \alpha_0} \delta_{\theta_k} + \frac{\alpha_0}{i - 1 + \alpha_0} G_0 \tag{2.14}$$

The distribution on partitions induced by the sequence of conditional distributions in Eq. 2.14 is commonly referred to as the *Chinese restaurant process*. The analogy is as follows. Consider a Chinese restaurant with an unbounded number of tables. Each ϕ_i corresponds to a customer who enters the restaurant, while the distinct values θ_k corresponds to the tables at which the customer sit. The i^{th} customer sits at the table indexed by θ_k, with probability proportional to n_k (in which case we set $\phi_i = \theta_k$), and sits at a new table with probability proportional to α_0 (set $\phi_i \sim G_0$).

Fig. 2.3 is an example of a Chinese restaurant (Teh et al., 2005).

2.2.2 Hierarchical Dirichlet Process

In order to share "share statistical strength" across different groups of data, the hierarchical Dirichlet process (HDP) (Teh et al., 2005) has been proposed to model the dependence among groups through sharing the same set of discrete parameters ("atoms"), and the mixture weights associated with different atoms are varied as a function of the data group (Ren et al., 2008).

A hierarchical Dirichlet process is a distribution over a set of random probability measures over $(\Theta, \mathbf{B})$. The process defines a set of random probability measures $(G_j)_{j=1}^{J}$, one for each group, and a global random probability measure G_0.

$$G_0|\gamma, H \sim DP(\gamma, H), \quad G_j|\alpha_0, G_0 \sim DP(\alpha_0, G_0) \tag{2.15}$$

Here, the global measure G_0 is distributed as a Dirichlet process with concentration parameter γ and base probability measure H and the random measures $(G_j)_{j=1}^{J}$ are conditionally independent given G_0, with distributions given by a Dirichlet process with base probability measure G_0.

The hyperparameter of the hierarchical Dirichlet process consist of the baseline probability measure H, and the concentration parameter γ and α_0. The baseline H provides the prior distribution for the parameter $(\phi_j)_{j=1}^{J}$. The distribution G_0 varies around the prior H, with the amount of variability governed by γ. The actual distribution G_j over the parameter θ_j in the j^{th} group deviates from G_0, with the amount of variability governed by α_0.

A hierarchical Dirichlet process can be used as the prior distribution over the factors for grouped data. For each j let $(\theta_{ji})_{i=1}^{n_j}$ be i.i.d. random variables distributed as G_j. Each θ_{ji} is a factor corresponding to a single observation x_{ji}. The likelihood

is given by:

$$\begin{aligned} \theta_{ji}|G_j &\sim G_j \\ x_{ji}|\theta_{ji} &\sim F(\theta_{ji}) \end{aligned} \tag{2.16}$$

Chinese Restaurant Franchise

Teh et al. (2005) have described the "Chinese restaurant franchise (CRF)" which is an analog of the Chinese restaurant process for hierarchical Dirichlet process. The CRF is comprised of J restaurants, each corresponding to an HDP group, and an infinite buffet line of dishes common to all restaurants. The process of seating customers at tables is restaurant specific.

Recall that the factors θ_{ji} are random variables with distribution G_j. $\{\theta_k|k = 1, ..., K\}$ denotes K random variables distributed according to H and for each j, $\psi_{j1}, ...\psi_{jT_j}$ denote T_j i.i.d. variables distributed according to G_0. Each ϕ_{ji} is associated with on ψ_{jt}, while each ψ_{jt} is associated with one θ_k. Let t_{ji} be the index of the ψ_{jt} associated with θ_{ji}, and let k_{jt} be the index of θ_k associated with ψ_{jt}. Let n_{jt} be the number of ϕ_{ji}'s associated with ϕ_{jt}, while m_{jk} is the number of ψ_{jt}'s associated with θ_k. Define $m_k = \sum_j m_{jk}$ as the number of ψ_{jt}'s associated with θ_k over all j.

From 2.14, the conditional distribution for θ_{ji} given $\theta_{j1}, ..., \theta_{ji-1}$ and G_0:

$$\phi_{ji}|\phi_{j1}, ..., \phi_{ji-1}\alpha_0, G_0 \sim \sum_{i=1}^{T_j} \frac{n_{jt}}{i - 1 + \alpha_0}\delta_{\psi_{jt}} + \frac{\alpha_0}{i - 1 + \alpha_0}G_0 \tag{2.17}$$

Since G_0 is distributed according to a Dirichlet process, G_0 can be integrated out by using Eq. 2.14 and writing the conditional distribution ψ_{jt} directly:

$$\psi_{jt}|\psi_{11}, \psi_{12}, ..., \psi_{jt-1}, \gamma, H \sim \sum_{k=1}^{K} \frac{m_k}{\sum_k m_k + \gamma}\delta_{\theta_k} + \frac{\gamma}{\sum_k m_k + \gamma}H. \tag{2.18}$$

This completes the description of the conditional distributions of the ϕ_{ji} variables. To obtain samples of ϕ_{ji}, first sample ϕ_{ji} using Eq. 2.17. If a new sample from G_0

is needed, Eq. 2.18 is used to obtain a new sample ψ_{jt} and set $\phi_{ji} = \psi_{jt}$. In the hierarchical Dirichlet process, the value of the factors are shared between the groups, as well as within the groups. This is a key property of hierarchical Dirichlet processes.

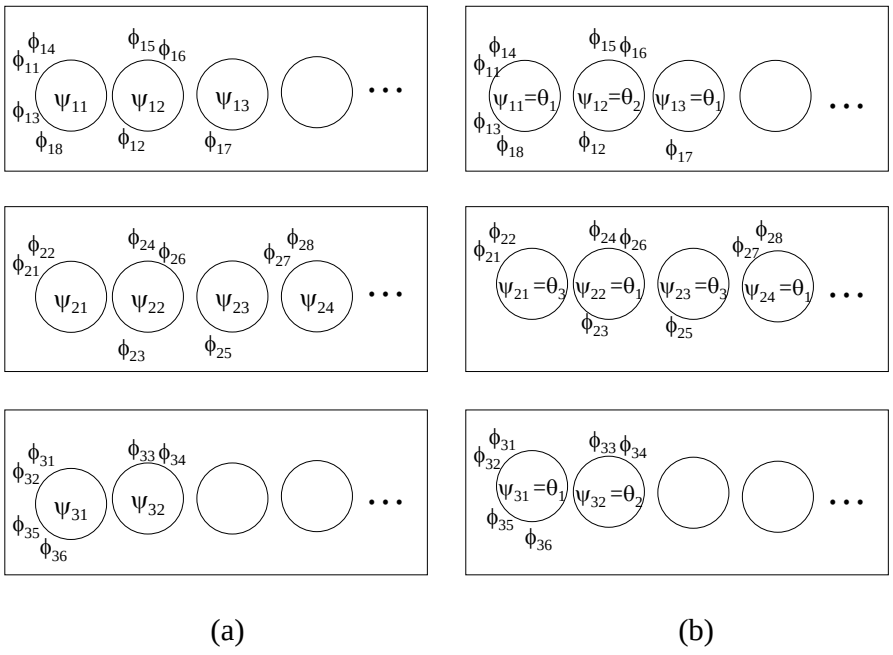


Figure 2.4: (a) A depiction of a hierarchical Dirichlet process as a Chinese restaurant. Each rectangle is a restaurant (group) with a number of tables. Each table is associated with a parameter ψ_{jt} which is distributed according to G_0, and each θ_{ji} sits at the table to which it has been assigned in Eq. 2.17. (b) Integrating our G_0, each ψ_{jt} is assigned some dish (mixture component) θ_k (image derived from (Teh et al., 2005)).

This generalized process is the Chinese restaurant franchise. The metaphor is as follows. There are franchise with J restaurants, with a shared menu across the

restaurants. At each table of each restaurant one dish is ordered from the menu by the first customer who sits there, and it is shared among all customers who sit at that table. Multiple tables at multiple restaurants can serve the same dish. The restaurant correspond to groups, the customers correspond to the ϕ_{ji} variables, the tables to the ψ_{jt} variables, and the dishes to the θ_k variables.

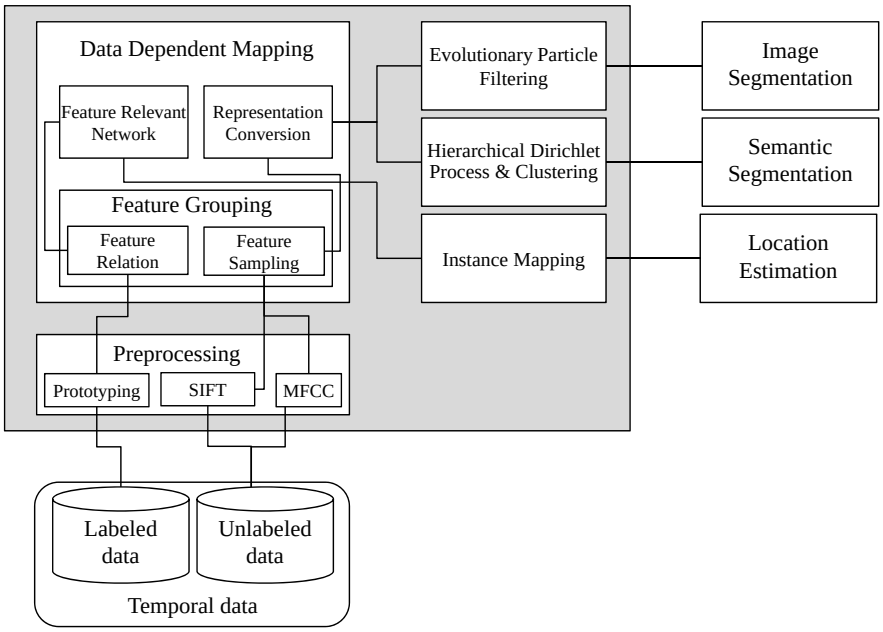


Figure 2.5: A combined framework in detail.

2.3 General Framework

In Fig. 2.5 and Algorithm 2, the general framework for individual works in this

Algorithm 2 Nonparametric analysis of temporal data

Input: $D = \{\mathbf{x}_{1:T}\}$

Begin

1. Constructing a feature relevance network

   1.1 Construct a new problem space based on feature relations.

   1.2 Map a new instance onto a problem space and compute the mapping cost.

   1.3 **Output**: a feature relevance network $\mathcal{Q}$

2. Segmenting a temporal stream

   2.0.1 Pre-processing of an image channel (**I**).

   2.0.2 Pre-processing of a sound channel (**S**).

   2.1 Non semantic analysis - for a unimodal channel (for **I** or **S**)

   2.1.1 Initialize a population of particle.

   2.1.2 While *not end of stream*

   2.1.2.1 Sample particles through resampling and genetic operators.

   2.1.2.2 Assign weight to each particle.

   2.1.2.3 Compute likelihood (L) of a new frame given the current population.

$$\begin{cases} L > \tau & \text{Go to step 2.1.2.1} \\ L < \tau & \text{Initialize a new population} \end{cases}$$

   2.2 Semantic analysis

   2.2.1 Estimate a HDP model for **I**.

   2.2.1.1 Compute likelihood of a new frame given the current HDP model.

   2.2.2 Estimate dialogue intervals in **S**.

   2.2.3 Dynamically merge changes in **I** and **S**.

   2.3 **Output**: a set of segment $\mathcal{S}$

thesis is explained. The proposed methods deal with temporal data such as video streams or Wi-Fi RSS (received signal strength) data. If a labelled data is given, then the given task is inference using a feature relevance network. If a video stream is given, appropriate segmentation would be selected based on the designated number of channels. Based on the given data, a feature relevance network learning or temporal stream learning is activated. A feature relevance network is a method which focuses on invariant relations among features. In order to infer in a unpredictably changing environment, a new problem space is built and data instances are mapped onto this new problem space. The mapping cost is interpreted as distance for classification. Chapter 3 provides more detailed explanation. For a sequential stream such as TV shows, it is possible to make segment in each unimodal channel or approximate semantic changes in a stream. For stream analysis, an image channel and a sound channel are pre-processed. If only an pre-processed image channel is designated, the pre-processed image channel is analyzed using Evolutionary Particle filtering. If both of image channel and sound channel are designated, the proposed method attempts semantic segmentation. With the given method, it is possible to select segments without prior assumptions on the underlying distributions or with more flexible assumptions based on nonparametric Bayesian framework. In Chapter 4 and Chapter 5, each approach is explained in more explicitly.

Chapter 3

Inference with Changing Underlying Distributions

In this chapter, we demonstrate how to deal with tasks in which underlying distributions are changing. A typical example of this domain is indoor location estimation problem (Yang et al., 2008). Wi-Fi data is noisy owing to the indoor environment's multipath and shadow fading effects. The data distribution changes constantly as people move and as temperature and humidity change. This is typical example of a non-parametric approach is required. In many cases, locations could be easily estimated using various traditional positioning methods and conventional machine learning approaches based on signalling devices, e.g., access points (APs). When there exist environmental changes, however, such traditional methods cannot be employed due to data distribution change. As a candidate solution, we introduce feature relevance network-based method which focuses on inter-relatedness among features.

3.1 Nonparametric Approach for Changing Distributions

Unlike other methods, our model is a non-parametric one making no assumptions about signal distributions. The proposed method is applied to the 2007 IEEE International Conference on Data Mining (ICDM) Data Mining Contest Task #2 (transfer learning) (Yang et al., 2008) which is a typical example situation where the training and test datasets have been gathered during different periods. As a result, the estimation framework obtained from traditional positioning methods such as TOA (time of arrival), TDOA (time difference of arrival), and RTOF (roundtrip time of flight) (Liu et al., 2007) cannot be used properly. In addition, conventional machine learning methods are not readily applicable due to the same reason (Caruana, 1997). As a result, the accuracy of the location estimation is worsened from 0.8227 (without distribution change) to 0.3223 (with distribution change).

Most previous approaches in these situations have tried to transform the parameters of statistically-learned models (Do and Ng, 2006; Pan and Yang, 2008). In the case of 2007 IEEE ICDM DMC Task #2, radio signal strength is not reliable and the number of available data instances from the changed distribution is too small to robustly transform the learned parameters. Therefore, it is nearly impossible to deploy parameter-transfer based approaches in a smooth manner.

Instead of employing parameters based approaches, we focused on inter-relatedness among features. Intuitively, a good feature representation is crucial for a successful domain adaptation (Ben-David et al., 2007). But distributional changes make it difficult to find a proper representation of features. Our method focuses on inter-feature relationship to construct plausible feature representation, which is expected to be resilient to distributional changes of feature values. The core assumption is that the nearer the access points are located, the more probable they are observed

simultaneously. Based on this expectation, we search for the access point (AP) pairs, highly adjacent to each other. Such AP pairs comprise edges of a graph structure and then, the problem space is reconstructed using it. When a new test instance is given, it is mapped onto the new problem space and expanded. More precisely, the test instance and the prototype of a class are expanded together until convergence. After that, the most plausible location is chosen. In the demanding task of 2007 IEEE ICDM DMC Task #2 where training and test datasets are obtained from different distributions and there are too few training instances from the test environment, our method shows superior results compared to those from the previous approaches. We achieve the accuracy of 0.5831 (upper bound) and 0.3238 (with the current setting, which is better than the best performance achievement ever).

The structure of this chapter is as follows. In Section 3.2, we review the related works. Section 3.3 explains the 2007 IEEE ICDM DMC problem and the proposed method. Section 3.4 presents experimental results. In Section 3.5, we discuss the characteristics of the proposed approach..

3.2 Related Works

Recently, transfer learning is receiving much attention. Transfer learning emphasizes knowledge transfer across domains, tasks, and distributions that are similar but not the same. In a survey paper (Pan and Yang, 2008), the authors summarized different settings of transfer learning as inductive, transductive, and unsupervised transfer learning. Inductive transfer learning aims to exploit unlabelled instances in classification tasks. In the self-taught learning framework, for example, the authors present an approach which learns a succinct, higher-level feature representation of inputs using unlabelled data (Raina et al., 2007). It is similar to our approach as it aims at finding a higher-level feature representation relying on both labelled and unlabelled

instances. However, we assume a underlying environmentally-invariant inter-feature relatedness and design new conceptual features based on this. In transductive transfer leaning setting, a lot of labelled instances in source domain are available while no labelled ones in target domain are available. In (III and Marcu, 2006), both in-domain and out-of-domain data are treated as if they are drawn from a mixture of distributions. In the location estimation problem, however, it is difficult to obtain such distribution due to the lack of data in target domain. In unsupervised transfer learning setting, researchers try to reduce the dimensionality with the help of related prior knowledge from other classes in the same type of concept (Wang et al., 2008). We search for useful features in expanded feature space without explicit prior knowledge.

Contrary to the previous researches, we focus on the representation. The method of explicitly minimizing difference between source and target domains in (Ben-David et al., 2007) seems similar to our approach. However, we concentrate on building feature relevance networks in order to infer missing features. If one uses parametric learning, methods in (Dai et al., 2007; Huang et al., 2007) would be useful. After estimating an initial distribution $\mathcal{D}_l$, the authors revise the model for a different distribution $\mathcal{D}_u$ of test data or produce re-sampling weights by matching the distributions between training and test sets in a feature space. We do not try to estimate parameters of an assumed distribution. As in (Jebara, 2004), our method builds a common representation space for multiple datasets in order to reinforce newly created feature space. In (Lee et al., 2007), the authors assume that features themselves have meta-features that are predictive of their relevance to the task, and model their relevance as a function of the meta-features using hyperparameters. The relationship among features regardless of the relevance to location is of primary importance to our method. Some researchers try to discover shared parameters or priors between

source and target domain modes (Lawrence and Platt, 2004; Evgeniou and Pontil, 2004). Because we do not assume a parametric model, these approaches have little relevance to our method.

The proposed method seeks artificial feature combinations suitable for a given problem. In terms of local consistency, this is similar to (Zhou et al., 2004). The difference is that our method does not assume a globally stable state, rather tries to figure out clusters of neighboring locations. A framework for predictive structures can be obtained from multiple tasks and there might be also a common framework for estimation location (Ando and Zhang, 2005). The idea of searching for feature similarity is also interesting, although it is uncertain how the idea could be applied on transfer learning.

Contrary to the traditional common feature representation space (Pekalska and Duin, 2008), we estimate expansion cost based on the assumption that the invariant relationship among features could be used as distance measure. In this sense, the proposed method could be interpreted as employing the manifold assumption (Belkin and Niyogi, 2001). Similar to (Cai et al., 2009), we derive local manifold structures from feature relevance networks and the distance between two locations is measured using these structures. Because given instances are compared to prototypes, our method is more similar to proximity-based representation spaces method, not to kernel methods.

As well explained in (Chen et al., 2010), existing methods for manifold learning have several shortcomings such as small sample size problem and information loss problem. Because the proposed method extracts inter-relatedness from observed instances regardless of their label, it is less limited by the size of labelled data. Our method is also similar to neighborhood detection in (Pan and Billings, 2008). By focusing on reconstruction of the local state vectors at some specified sites from

measured data, the authors try to determine the spatiotemporal neighboring region or sites. We propose a method to determine the necessary dimensions of the reconstruction vector, although we do not consider the temporal characteristics in data. In (Chow et al., 2008), the idea of using frequency for intracluster similarity and intercluster difference is similar to ours. However, we do not preselect the number of clusters.

The indoor location estimation has been researched based on various technologies. In (Gwon et al., 2004), the idea of forming the region of confidence (RoC) is similar to our method but the authors obtain the RoC using geometric properties. Our method does not consider the geometric properties and is solely based on feature relatedness. In (Pan et al., 2005), the authors address the location estimation in an 802.11 wireless LAN environment. The authors perform a kernel-based transformation of signal and physical spaces to capture the nonlinear relationship between signals and locations. This approach does not provide explicit solutions for transfer learning. In (Yin and Yang, 2008), the authors address a problem similar to the task of this paper. A radio map is constructed by calibrating signal-strength values in offline phase. However, this method uses the luxury of neighboring reference points not available in our problem setting.

3.3 Feature Relevance Network Learning

3.3.1 Indoor Location Estimation Problem

The 2007 Data Mining Contest, sponsored by the IEEE International Conference on Data Mining, provides the first realistic public benchmark dataset for indoor location estimation using radio signal strength (RSS) that a client device receives from a set of Wi-Fi APs. The dataset was collected from an 145.5 m $\times$ 37.5 m academic

building at the Hong Kong University of Science and Technology where the location is divided into a grid of 247 units. The contest focused on two tasks: indoor location estimation and transferring knowledge for indoor location estimation. We focus on transferring knowledge for the indoor location estimation task.

Let $\mathcal{Z}$ ($|\mathcal{Z}| = 3,128$) be the set of test instances. Each point $\mathbf{z}_i$ ($\mathbf{z}_i \in \mathcal{Z}$) is expected to belong to one of M locations ($M = 247$). We will use c_i to denote the location associated to $\mathbf{z}_i$. Our goal is to estimate c_i for $\mathbf{z}_i$. Basically, the contest is to predict locations on the basis of RSS values received from the Wi-Fi APs. In this paper, training instances are denoted as $\mathcal{T}$ ($\mathcal{T}= \mathcal{T}_L \cup \mathcal{T}_U$). $\mathcal{T}_L$ denotes labelled instances ($|\mathcal{T}_L| = 621$), and $\mathcal{T}_U$ denotes unlabelled instances ($|\mathcal{T}_U| = 1,701$). Each $\mathbf{t}_l$ ($\mathbf{t}_l \in \mathcal{T}_L$) is denoted as $\mathbf{t}_l = \langle \mathbf{a}_l, c_l \rangle$ and $\mathbf{a}_l = \langle a_{l,1}, ..., a_{l,99} \rangle$ ($a_{l,i}$ means the RSS value from the ith access point of the lth instance). Each $\mathbf{t}_u$ ($\mathbf{t}_u \in \mathcal{T}_U$) is represented as $\mathbf{t}_u = \langle \mathbf{a}_u \rangle$ and $\mathbf{a}_u = \langle a_{u,1}, ..., a_{u,99} \rangle$. The distributions of $\mathcal{Z}$ and $\mathcal{T}$ are different, so the traditional assumption that training and test examples are sampled from the same distribution does not hold in this task. A few reference points with location label from the test environment are provided as landmarks, i.e., $\mathcal{D}$ ($|\mathcal{D}| = 52$), to assist the estimation procedure. Difficulties of the estimation problem come from the following two facts. The first is that properties of some locations are too similar. The second is that the given landmark instances are scarce. Due to the lack in landmark data, one cannot build a reliable parameter transfer model for each location.

3.3.2 The Proposed Method

Fig. 3.1 illustrates the overall algorithm. The proposed method is composed of mapping, structure expansion, and estimation steps. In the mapping step, the feature relevance network, $\mathcal{Q}$, is obtained from $\mathcal{T}$. $\mathcal{Q}$ consists of AP pairs supposed to be adjacent. The adjacency of APs is inferred based on their co-occurrence. In addi-

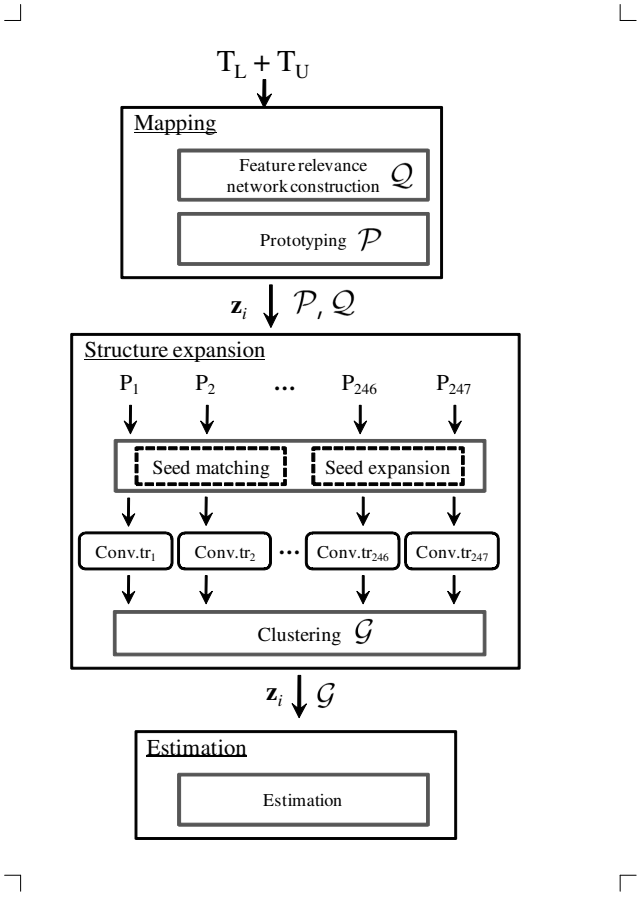


Figure 3.1: **Overview of the proposed method**. The proposed method is composed of mapping, structure expansion, and estimation steps. At the mapping step, environmentally invariant properties are obtained. The feature relevance network $\mathcal{Q}$ is constructed from $\mathcal{T}$. P_r ($P_r \in P$), the prototype of location r is computed from $\mathcal{T}_L$. At the structure expansion step, a test instance and P_r are expanded until converging to the same structure based on $\mathcal{Q}$. $Conv.tr_i$ denotes the ith converged tree. Based on the accumulated cost of the expansion, some locations are selected as a member of the neighboring group, $\mathcal{G}$. At the estimation step, the location for the given test instance is estimated from $\mathcal{G}$.

tion, the prototype for each location ($\mathcal{P} = \{P_1, ..., P_{247}\}$) is obtained using $\mathcal{T}_L$. In the structure expansion step, each prototype and a test instance are expanded until they are converged to the same structure. At the estimation step, c_i is estimated from a cluster of neighboring locations ($\mathcal{G}$).

Mapping

The aim of this step is to find highly related AP pairs using the procedure in Table 3.1. At first, the feature relevance network $\mathcal{Q}$ is obtained from $\mathcal{T}$. $\mathcal{Q}$ is defined as a triple $\mathcal{Q} = (\mathbf{X}, \mathbf{E}, \mathbf{W})$, where $\mathbf{X}$=$\{\mathbf{x}_1, ..., \mathbf{x}_{|X|}\}$, $\mathbf{E}$=$\{\mathbf{E}_1, ..., \mathbf{E}_{|E|}\}$, $\mathbf{W}$=$\{\mathbf{w}_1, ..., \mathbf{w}_{|E|}\}$ are the sets of vertices, edges, and weights, respectively ($\mathbf{x}_i$ means the ith AP, $|X|$ is the number of APs, and $|E|$ denotes the number of AP pairs, i.e., $4,851$). $\mathbf{W}$ denotes the relevance between two APs. $\mathbf{W}$ is defined using Eq. 3.1. If two APs are not observed concurrently, then $AValue$ is a predefined NULL value. Otherwise, $AValue$ is defined as follows.

Definition 1 Association value, $AValue_{j,k}$ of an edge, $E_{j,k}$ between APs j and k is defined as,

$$AValue_{j,k} = 1 - \frac{Freq_{j,k}}{\mathrm{MAX}_{Freq} + 1}, \tag{3.1}$$

where $Freq_{j,k}$: frequency of the cases that AP j and k are observed concurrently, MAX_{Freq}: maximum frequency. $AValue$ is computed from $\mathcal{T}$.

In the mapping step, the set of prototypes, $\mathcal{P}$, representing each location is also constructed from $\mathcal{T}_L$ as follows:

For all $\mathbf{t}_l$ whose location is r,

$$P_r = \langle \bar{\mathbf{a}}_r, r \rangle, \tag{3.2}$$

Table 3.1: Mapping into a new problem space

Step 1 Generate all possible AP pairs from the 99 APs.

($E_{j,k}$: j=1,...,98, k=$j+1$,...,99)

For each $E_{j,k}$

Step 2 $Freq_{j,k} = 0$.

Step 3 Count frequency $Freq_{j,k}$ from $\mathcal{T}$.

$Freq_{j,k} = Freq_{j,k} + 1$,

if AP_j and AP_k are observed concurrently.

Step 4 Calculate association value,

$AValue_{j,k}$ for all AP pairs.

End of loop

For each location

Step 5 Make the prototype, P_r, for each location r using $\mathcal{T}_L$.

End of loop

where $\bar{\mathbf{a}}_r = \langle \bar{a}_{r,1}, ..., \bar{a}_{r,99} \rangle$ ($\bar{a}_{r,h}$ is the averaged RSS value of the hth AP of the instances whose location is r).

Structure expansion

The aim of this step is to form the candidate location set $\mathcal{G}$ for a given test instance. $\mathcal{G}$ corresponds to a group of locations adjacent to the given test instance. In order to select the candidate locations, the test instance and the ith location's prototype are compared and expanded until convergence. The expansion cost is interpreted

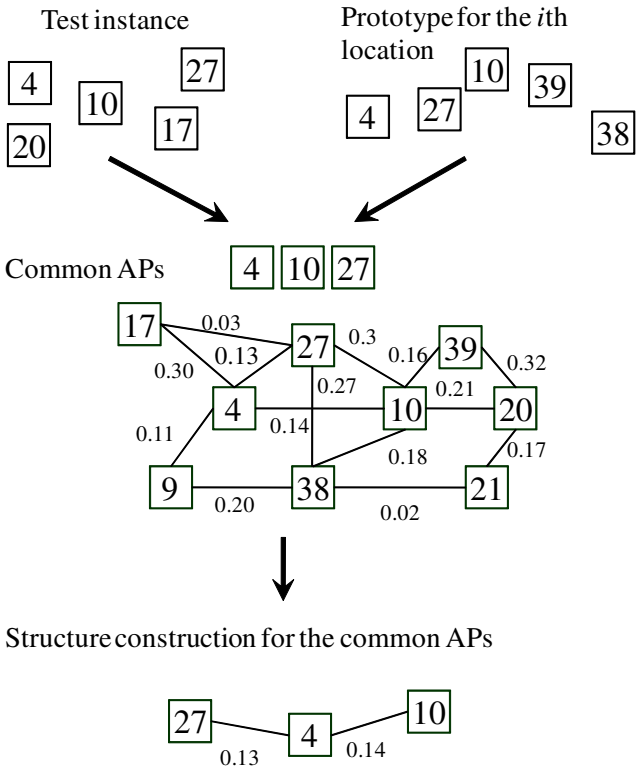


Figure 3.2: **Illustration of seed matching**. At this step, the common APs observed in a test instance and the prototype of the ith location are processed. In the figure, AP_4, AP_{10}, and AP_{27} compose S_{COM}. Based on the $AValue$s, the edges with the lowest $EValue$, $E_{4,27}$ and $E_{4,10}$ are selected and comprise the connecting structure S_{COM}.

as distance. The structure expansion step is composed of *seed matching* and *seed expansion*.

Seed matching procedure is summarized in Table 3.2. For a test instance $\mathbf{z}_i$ and a prototype P_r, the commonly observed APs in $\mathbf{z}_i$ and P_r constitute the set

Table 3.2: Seed matching and expansion

Step 1 Form S_{COM} (set of commonly observed APs), S_{PL} (set of APs existing only in P_r and not in $\mathbf{z}_i$), S_{TL} (set of APs existing only in $\mathbf{z}_i$ and not in P_r) from test instance $\mathbf{z}_i$ and location prototype P_r.
Step 2 (**Seed matching**) Obtain set of $E_{j,k}$'s (E_{COM}) connecting the elements of S_{COM} based on $AValue_{j,k}$ using an MST algorithm.
Step 3 (**Seed expansion**) Obtain set of $E_{j,k}$'s (E_{EXP}) connecting S_{COM}, S_{TL}, S_{PL} based on $AValue_{j,k}$ using an MST algorithm.

of common APs (S_{COM}). A structure connecting the APs in S_{COM} is constructed using a minimum spanning tree (MST) algorithm (Cormen et al., 2001). $EValue$ for this structure is computed as follows:

$$EValue_r = \sum_{E_{j,k} \in E_{COM}} AValue_{j,k}, \tag{3.3}$$

where E_{COM} is the structure for S_{COM} obtained by an MST algorithm.

In the example illustrated in Fig. 3.2, the observed APs of a test instance are $\{4, 10, 17, 20, 27\}$, the observed APs of a prototype are $\{4, 10, 27, 38, 39\}$, and S_{COM} is $\{4, 10, 27\}$. Then, edges with lower $AValue$ are selected. Therefore, $E_{COM} = \{E_{4,10}, E_{4,27}\}$.

At the seed expansion step, a test instance and the location's prototype are expanded by adding unobserved APs until convergence. The aim of this step is to

find a set of edges, E_{EXP} connecting APs in S_{TL}, S_{PL}, and S_{COM} (where S_{TL} is the set of APs observed only in a test instance. S_{PL} is the set of APs observed only in the ith prototype) using the procedure summarized in Table 3.2. This step is also based on an MST algorithm. The edges in E_{EXP} are used to compute $LValue$. This is based on the assumption that if an AP is not observed due to an environmental change, it could be guessed from nearby APs using the feature relevance network.

$$LValue_r = \sum_{E_{j,k} \in E_{EXP}} AValue_{j,k}. \tag{3.4}$$

In Fig. 3.3, the seed expansion step is illustrated using the same example in Fig. 3.2, where $S_{COM} = \{4, 10, 27\}$, $S_{TL} = \{17, 20\}$, and $S_{PL} = \{38, 39\}$. At the seed expansion step, a structure connecting APs $\{4, 10, 17, 20, 27, 38, 39\}$ is constructed by adding a new AP '9'. After the execution of the MST algorithm, $E_{EXP} = \{E_{9,38}, E_{4,9}, E_{4,27}, E_{4,10}, E_{4,17}, E_{10,39}, E_{10,20}\}$.

After the structure expansion step, $EValue$ and $LValue$ are obtained for each pair of the test instance and the ith location's prototype. $EValue$ represents the degree of relatedness between the test instance and the prototype. $LValue$ corresponds to the accumulated expansion cost. Because two observations obtained at different periods at the same location would have more common APs and lower expansion cost than observations from different locations, the ratio of these two values can be used as a distance. This idea is represented as follows:

$$RScore_r = \frac{LValue_r}{EValue_r} \quad (r = 1, ..., 247). \tag{3.5}$$

Now, the locations supposed to be located more adjacent to the given test instance can be recommended based on $RScore$ values (Eq. 3.5). These locations will be denoted as $\mathcal{G}$. The other locations will be represented as $\mathcal{G}'$. $\mathcal{G}$ and $\mathcal{G}'$ can be obtained after sorting the 247 prototypes according to the their $RScore$s, then dividing them

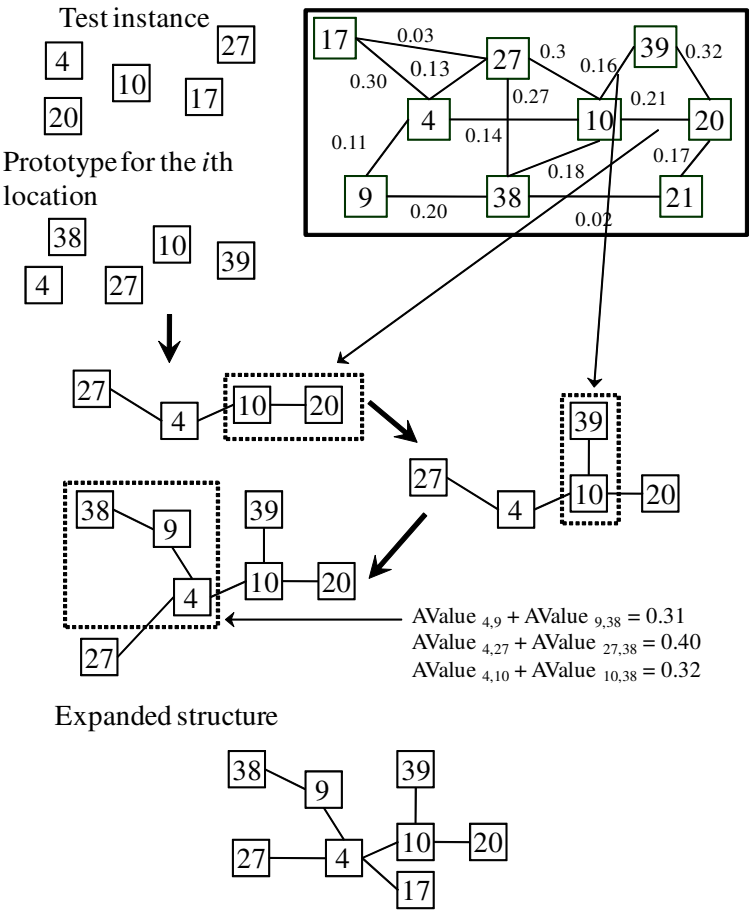


Figure 3.3: **Exemplary seed expansion procedure for missing APs**. E_{EXP} connecting APs in a test instance and the rth prototype is constructed by adding a new AP '9'.

Table 3.3: Recommendation

For whole *RScore* values of a given test instance $\mathbf{z}_i$

Step 1 Compute Eq. 3.6 for all *RScore* values.

Step 2 Compute ψ values based on Eq. 3.7.

Step 3 Select set of interim criteria $\eta_{interim}$ based on ψ values showing significant variation.

Step 4 Compute $F_{INC} = \sum_{k=1}^{|D|} I_k$ by classifying landmark data based on $\mathcal{G}$ for all $\eta_{interim}$.
(if the kth landmark instance is classified correctly, $I_k = 1$ else $I_k = 0$)

Step 5 Select the final criterion
$\eta_{final} = \text{argmax}_{\eta_{interim}} F_{INC}$

by a proper criterion.

In order to select the proper criterion, we devise two operators γ and ψ as follows:

$$\gamma_r = RScore_r - RScore_{r-1}. \tag{3.6}$$

$$\psi_r = \left\{ \begin{array}{ll} \frac{\gamma_{r+1}}{\gamma_r} & \text{if } \gamma_{r+1} > \gamma_r \\ \frac{\gamma_r}{\gamma_{r+1}} & \text{otherwise} \end{array} \right\}. \tag{3.7}$$

If one selects too many candidates for $\mathcal{G}$ in order to contain c_i for the given test instance, the estimation accuracy would be worsened. On the contrary, if one selects too few candidates for $\mathcal{G}$, then the estimation accuracy would also be worsened because c_i would not be included in $\mathcal{G}$. A criterion capable of balancing the number of elements in $\mathcal{G}$ is searched using the procedure in Table 3.3.

Figs. 3.4 and 3.5 show $RScore$s and ψ values for some test instances. As shown in Fig. 3.4, it is difficult to find a meaningful variation in the $RScore$ curves. However, the curves in Fig. 3.5 show prominent changes in ψ values. A set of interim criteria is selected based on the fluctuations in ψ curves. For the landmark instances, a set of interim criteria, $\eta_{interim}$, is selected based on these ψ values. Each element of $\eta_{interim}$ is determined as ψ_l where $\psi_l \geq \psi_{init}$ (ψ_{init} is a constant determined using landmark instances). For each element in $\eta_{interim}$, η_{final} is a value maximizing F_{INC} in Table 3.3.

After constructing $\mathcal{G}$, RSS values of some prototypes are modified. If a location is adjacent to a known location (i.e., landmark data), the amount of RSS value change of the location is likely to be similar to the RSS value change of the known location. This idea is incorporated to adjust the RSS values of the APs of the prototypes in $\mathcal{G}$.

Estimation

The location of a test instance is estimated at this step based on the weighted p-norm distance. For a test instance $\mathbf{z}_i$ and $\mathcal{G}$, location $c_i{}^*$ for $\mathbf{z}_i$ is estimated as follows:

$$c_i^* \leftarrow \text{argmin}_{r \in \mathcal{G}} dist(P_r, \mathbf{z}_i). \tag{3.8}$$

$$dist(P_r, \mathbf{z}_i) = \sum_{d=1}^{99} (w_d \cdot |a_{r,d} - z_{i,d}|^p)^{\frac{1}{p}}, \tag{3.9}$$

where $p = 1.7$ (p was optimized using landmark instances), $w_d = \frac{|a_{r,d} - z_{i,d}|}{g}$ (if $a_{r,d}$ or $z_{i,d}$ is unobserved, then $w_d = 0$, $g = \sum_{d=1}^{99} z_{i,d}$ for the observed $z'_{i,d}s$).

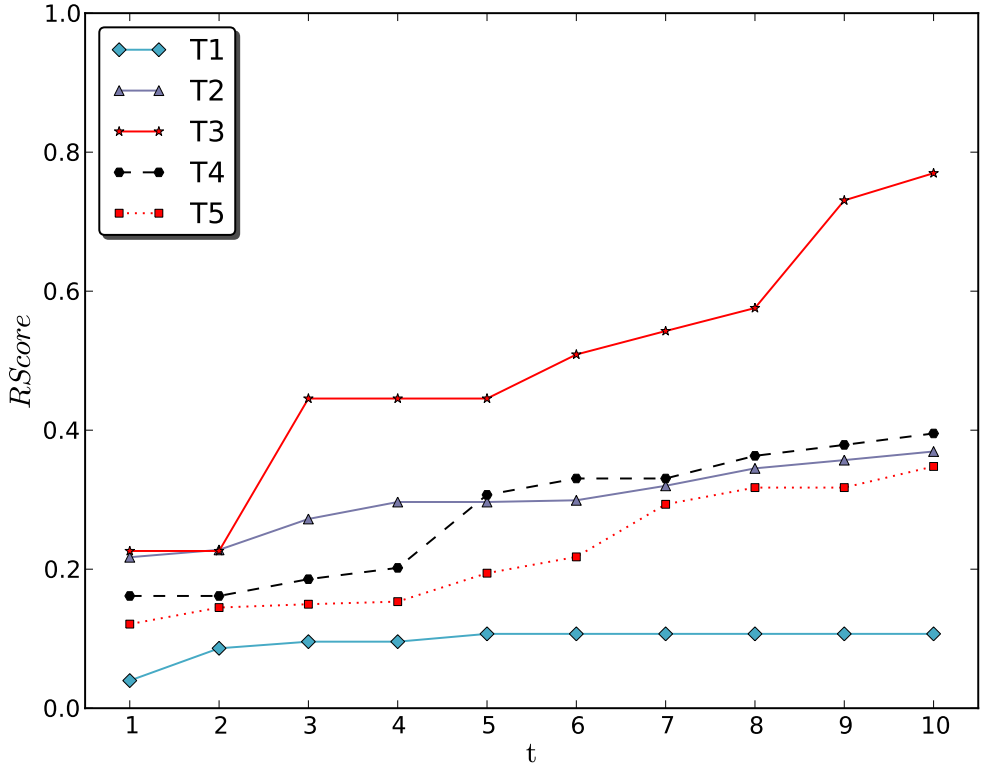


Figure 3.4: **RScore curves**. This chart shows variations in *RScores* of five test instances (T1, T2,...,T5). There hardly seems a clear cut point. Here, t means ten prototypes.

3.4 Experimental Results

In Fig. 3.6, the feature relevance network built from $\mathcal{T}$ is shown. Here, linked APs are simultaneously observed pairs and they are assumed to be adjacent. Interestingly, we could observe some hub APs and conjecture that the feature relevance network

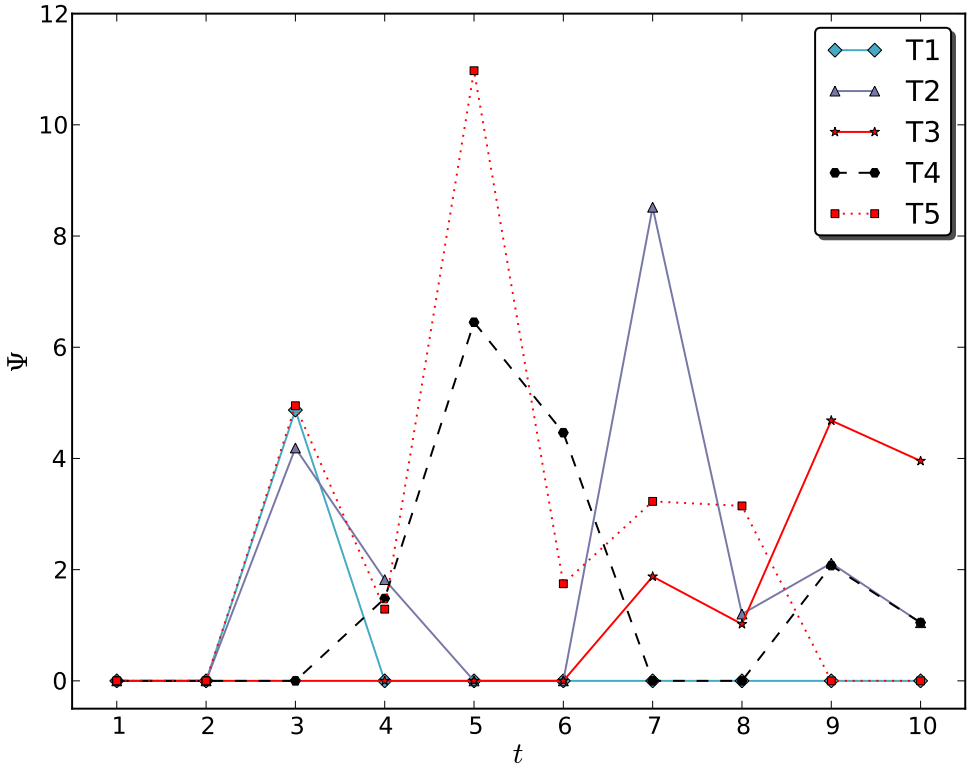


Figure 3.5: ψ **curves for Fig. 3.4**. This figure shows ψ curves for the instances in Fig. 3.4. By introducing γ and ψ measures, there are fluctuations in the curves. A set of interim criteria is selected based on the fluctuations. If the denominator is 0, the corresponding ψ value is defined as 0.

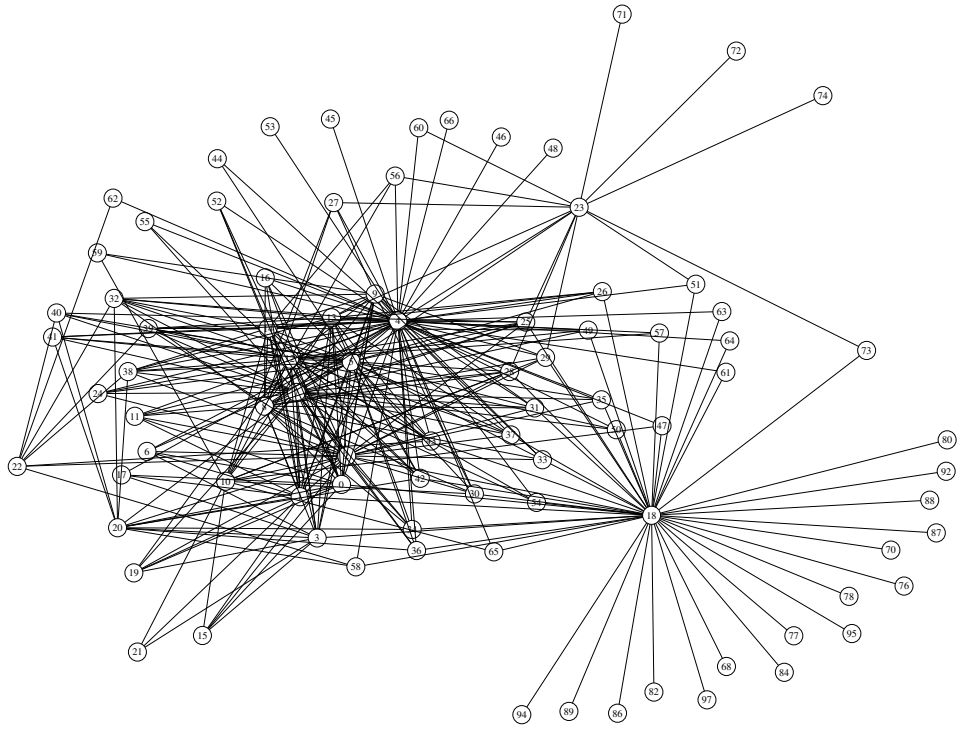


Figure 3.6: **The generated relevance network**. This figure displays the relatedness between the 99 APs. Linked APs are more likely to be adjacent to each other than other pairs. Here, weights or adjacencies are not shown. These links represent all the concurrency among the APs estimated from $\mathcal{T}$.

assumes a complex network structure whose degree distribution can be represented by the power law.

3.4.1 Quality of Candidate Recommendation

After the recommendation, the cluster $\mathcal{G}$ for a $\mathbf{z}_i$ is obtained. An example $\mathcal{G}$ for a test case where $c_i = 203$ is illustrated in Fig. 3.7. As shown in Fig. 3.7, the recommended locations are likely to be adjacent to the target location. For more

Table 3.4: Quality of candidate recommendation

Analysis	*Value*
Average AD_i	2.671
Hit ratio (%)	69.7%
Maximum size of a cluster	10
Minimum size of a cluster	4
Average size of a cluster	6.773

comprehensive analysis, we present two concepts describing the degree of adjacency (AD) and hit ratio respectively. The degree of adjacency corresponds to the density of $\mathcal{G}$, i.e., the average distance between the locations in $\mathcal{G}$. It was calculated using the offered location map. We assign coordinates to each location by setting the left-bottom location as $(0, 0)$ and the distance between adjacent locations as one. This map was *only* used for computing the degree of adjacency and had *not been used* in the learning and estimation processes. Then, the degree of adjacency, AD_i of the ith location is computed using the following equation.

$$AD_i = \frac{\sqrt{\sum_k (\mathbf{x}_k - o_i)^2}}{\#\ of\ elements\ in\ \mathcal{G}}. \tag{3.10}$$

(o_i: the (artificial) center point of $\mathcal{G}$, $\mathbf{x}_k$: the kth element in $\mathcal{G}$).

The degree of adjacency denotes the quality of a cluster. The higher is AD, the worse is the quality of $\mathcal{G}$. However, even low AD values could be useless if actual c_i is not included in $\mathcal{G}$. Hit ratio means the ratio of $\mathcal{G}$ containing actual c_i of each $\mathbf{z}_i$. With higher hit ratio, it is possible to obtain more higher accuracy.

The maximum size (it means the number of locations in $\mathcal{G}$) of $\mathcal{G}$ is 10. The

minimum size is 4. The average size of a cluster is 6.773. As explained in Table 3.4, the recommendation capability is satisfiable. In other words, 69.7% of each c_i is contained in the recommended cluster, $\mathcal{G}$. Furthermore, the recommended locations are actually adjacent ones according to the average AD value of 2.671.

3.4.2 Prediction Performance and Comparisons

In order to evaluate the prediction performance, we carry out two experiments. In the ideal setting, the group of optimal adjacent locations, $\mathcal{G}^o$, is determined through reference to the actual location of each test instance in order to show the upper bound in the ideal case with the given condition. In the other case, $\mathcal{G}$ is determined based on the procedure in Table 3.3.

In each case, the accuracy is determined by the following definition

.

Definition 2

$$Accuracy = \frac{\sharp\ of\ correct\ predictions}{\sharp\ of\ all\ test\ instances}.$$

In Table 3.5, the estimation results are presented. With the proposed algorithm, the adjacent locations, $\mathcal{G}$ for each $\mathbf{z}_i$, are recommended during the recommendation step. Therefore, instances with similar feature vectors are given to a classifier. As a result, it is possible to achieve relatively higher accuracy with simple classifier. For 69.7% of hit ratio, the accuracy is 0.3238. This performance is better than the first ranked result of the 2007 IEEE ICDM DMC Task #2 and the conventional machine learning algorithms such as neural networks, kNN, random forest, and naive Bayes (algorithms implemented in WEKA 3.4.11 have been employed for the comparison). The broadly used algorithms do not provide promising results. Because other classifiers do not have the transfer learning capability, the poor performance is expected. These results are presented to show that the given task cannot be solved well with

conventional methods.

The winners of the ICMD07 contest employed the Minkowski distance and NUM (nearest unlike neighbor) distance. The winners computed the initial class centers and assigned a class label to the unlabelled instances. After that, new class centers were computed using the newly labelled instance and the given labelled training instances. Finally, the Minkowski distance weights for each class were assigned. Contrary to the ICDM07 contest winner, our method focuses on relational properties among APs. The relational properties are represented as relevance network. For a given test instance, candidate locations for this test instance are selected based on feature relevance network. Because the proposed method infers the likelihood of a feature change, our method can be applied to other domains with missing or changing attributes. This is one novelty of our method.

In order to present the upper limit of the proposed method in the current setting (with the current relevance network), we built an optimal cluster $\mathcal{G}^o$ which always includes the actual location of $\mathbf{z}_i$. In this case, the accuracy is 0.5831. The difference between the ideal case and the other case is due to the recommendation rule. The estimation accuracy in the case of using Table 3.3 can be improved further by calibrating the recommendation rule. With the current experimental setting, we have only 48 samples (discarding repeated locations) for estimating the characteristics of the test environment. With cues corresponding 19.4% of the target locations, there is much room for the improvement. By calibrating the relevance network with more instances, it is possible to improve the accuracy to the upper bound.

In order to verify the validity of the estimation after recommendation, we also compared the accuracy at each stage. The accuracy with *recommendation rule only* means the case where the location with the lowest $RScore_i$ is determined as c_i for the given $\mathbf{z}_i$. The accuracy with *classifier only* means the case where the classifier

Table 3.5: Experimental results. *Rank*1 and *Rank*2 are the best and the second estimation results at 2007 IEEE ICDM DMC contest (ICDM-web)

ID	*Accuracy*
Proposed method (upper bound)	0.5831
Proposed method (using Table 3.3)	0.3238
*Rank*1	0.3223
*Rank*2	0.3149
Naive Bayes	0.1992
Recommendation rule only	0.191
Random forest	0.1781
kNN (k=1)	0.1771
kNN (k=2)	0.1640
kNN (k=3)	0.1550
Neural networks	0.1544
Classifier only	0.0

of Eq. 3.9 is used for estimation without forming $\mathcal{G}$. Because the distance measure in Eq. 3.9 ignores unobserved APs, *classifier only* estimation result is inferior to other conventional classifier. However, our method improves the estimation accuracy significantly by combining the recommendation procedure and a simple distance measure.

A famous indoor location measurement method is *location fingerprinting*. In the location fingerprinting technique, the location fingerprints are collected by performing a site-survey of the received signal strength from multiple access points. The

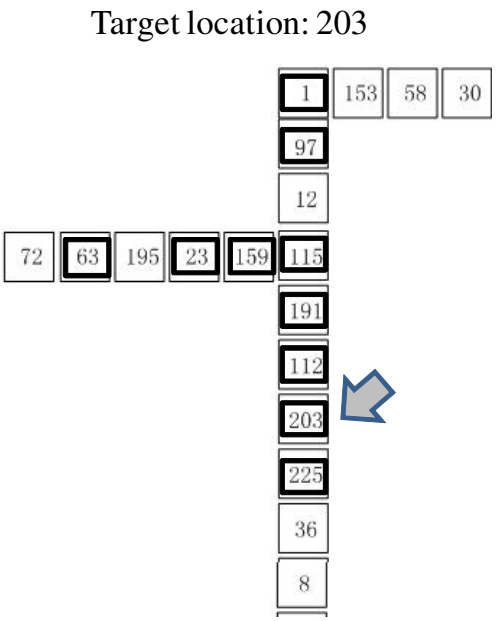


Figure 3.7: **Recommended locations for a test instance whose real location is** 203. For location 203, locations $112, 191, 115, 225, 203, 159, 63, 97, 1$, and 23 are recommended by our method.

RSS is measured with enough statistics to create a database or a table of predetermined RSS values on the points (Kaemarungsi and Krishnamurthy, 2004). Because location fingerprinting uses fingerprints of a location, it requires sufficient measurements of every location. In the given task of transfer learning, we cannot satisfy this requirement. Therefore, we develop a feature relevance network model.

3.5 Discussion and summarization

In spite of the difficulties of the indoor location estimation problem, the proposed method achieves meaningful improvement. The idea behind the improvements is as follows: The difficulty due to distribution change could be overcome by building environmentally-invariant properties. We represented co-occurrences among APs as a simple graph structure. Using this graph space, we obtained a method enabling location estimation despite the distribution change. The difficulty due to too few training examples was circumvented by giving up construction of location models. We could not build reliable location models with the given scarce labelled training data. Furthermore, even if there exist location models, we could not transfer the parameters due to the lack of labelled training data from the test environment. By focusing on feature relatedness, we got rid of the need for a model per a location.

The problem of insufficient labelled data always harasses researchers. In the proposed method, the effect of insufficient labelled data is a decline of the quality of prototypes. As a consequence, the seed matching/expansion steps are deteriorated. However, the proposed method has a unique strength in transfer learning. Our method focuses on the invariant relationship among features. Therefore, one could extract more meaningful invariant relations with plentiful unlabelled data and these upgraded relations result in a more informative relevance network. As a result, plentiful unlabelled data can offset the deteriorating effect of insufficient labelled data. With more unlabelled data from the test environment, we could secure another cue for estimating the RSS value change. Therefore it is possible to enhance the estimation result further.

Another virtue of the proposed method is in its recommendation step. Through recommendation, we formed a cluster of adjacent locations. By focusing on these neighboring locations, we were able to estimate the unknown location more success-

fully. Our method is suitable for such tasks in which the number of classes are very large, the independency among classes is weak, and the number of training instances per class is small. The indoor location estimation problem is a typical example for this kind of domain, thus we succeeded in introducing a useful estimation method.

One of shortcomings of the proposed method is its computational cost. The proposed method builds a relevance matrix of fixed size $F \times F$ (F: the number of features). Because the relevance network is obtained from co-observation of APs, the large size of training data is not a serious computational burden. The heaviest computational burden comes from the process of fixing ψ values based on $RScore$ values. Although the running time for determining ψ values is very high (as shown in Fig. 3.8), there is a margin for further optimization. The running time is proportional to the number of landmark instances and the computation time of each ψ value could be improved with more instances. In the employed MST algorithm, the running time per location is $l \times O(m\log n)$ (l: the number of locations, i.e., 247, m: the number of APs, n: the number of edges, i.e., 4851). In the equation $O(m\log n)$, m is fixed and n can be reduced by employing some dimensionality reduction techniques such as MDS (multi-dimensional clustering). With enough data showing distribution change (labelled or unlabelled ones), it is possible to eliminate less informative elements from the the relevance network based on the co-occurrence. In addition, more informative prototypes (represented by less features) could be obtained and these refined prototypes could contribute to pruning the edges. As a result, n in $O(m\log n)$ could become smaller and computational burden could be relieved given a sufficient number of informative instances.

A feature relevance network represents how closely two features are related to each other. Therefore, the proposed method can be extended to other problems if a few preconditions are met. The preconditions are as follows: 1) there should

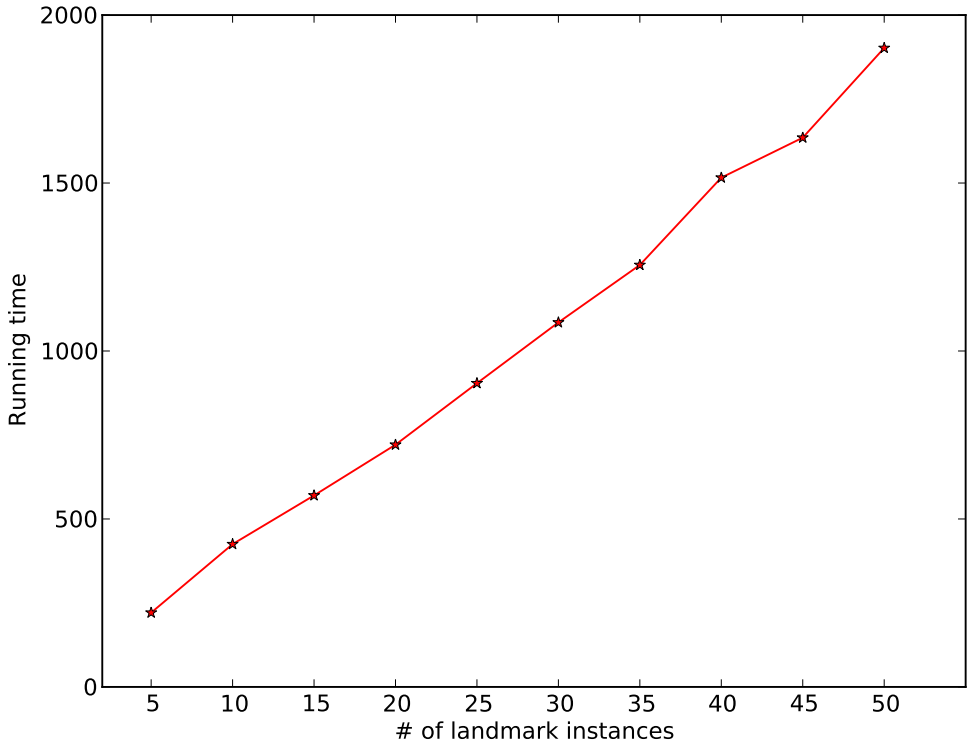


Figure 3.8: **Running time for determining ψ value**. This chart shows the running time along the number of landmark instances. The time is measured in seconds. The running environment: Intel core 2 CPU 6600 2.40 GHz and 2.39 GHz with 2.00GB RAM.

be a reasonable basis to support the assumption that features are interrelated in a pairwise manner - in the case of indoor location estimation, it is highly probable that the property of adjacent APs would be similar; 2) the unvaried relation among features is easy to guess or represent - in the case of indoor location estimation, adjacent APs can easily be represented by their observation values, which should be highly correlated to one another. There could be lots of domains satisfying these preconditions. Transfer learning of object classes (Heitz et al., 2005) is a typical example. In the object recognition domain, the contour corresponds to a AP in indoor location estimation and the relations among contours are invariant. In this case, the relations among contours can be represented as a feature relevance network. As a result, it is possible to recognize an object even some contours are missing based on the relevance network.

Chapter 4

Temporal Stream Learning with Evolutionary Particle Filtering

In this chapter, we demonstrate how to perform segmentation in sequential streams where the number of changepoints is unknown. Although previous temporal models report promising results, they have focused on sophisticated prior assumptions and thus not appropriate for real-world streams such as TV dramas. For more flexible processing, this approach is based on the use of particles and evolution. Contrary to conventional particle filtering, the proposed scheme represents a segment via a population of particles in a collaborative manner. This collaborative representation makes the proposed method suitable for sequential streams because a population based scheme can tolerate minor distortions in a stream using dominant patterns retained in a population. In order to introduce essential diversity into a population, genetic operators are employed. Sequential dependencies in a stream is learned by estimating transitional probability between segments. Our approach can be useful, for example in a stream summarization. The proposed method is evaluated on three episodes of a TV drama and qualitative/quantitative results indicate that our

method provides a useful stream representation method with high detection rates.

4.1 Collaborative Particles and Temporal Stream Analysis

Research on a temporal domain has been actively performed. Among various tasks, detecting regions of change in a stream is of widespread interest due to large number of applications in diverse disciplines (Radke et al., 2005). Previous approaches for sequence learning have focused on changes in low-level features such as pixel, block-based diagram, or histogram comparison (Koprinska and Carrato, 2001). A stimuli-driven learning process governed by the principle of self-organization (Barreto and Araújo, 1999) or a sequence learning by adaptively combining a set of reusable primitives (Namikawa and Tani, 2008) were also introduced. However, these approaches are unappropriate for real-word data sets because low-level features are not robust enough and repeating frames are too inflexible.

In this chapter, we propose a novel hidden dependency learning scheme for temporal sequences such as TV dramas. The proposed method attempts to represent a temporal stream using a set of particles and a transitional probability matrix based on evolutionary particle filtering (PF) (Kwok et al., 2005; Park et al., 2009). In this task, the important task is to distinguish states for inferring transitional probability. However, the practical difficulty is that a portion of dominant characteristics are changed irregularly due to camera works or lighting. We alleviate this irregular distortion by representing states via a population of particles in a collaborative manner. In the proposed scheme, it is possible to tolerate minor distortions because dominant patterns are retained in a population. This method consists of two-stage of evolution and dependency learning. At the evolution stage, characteristics of dominant features are extracted though evolution and these characteristics are uti-

lized for segmentation. At the dependency learning stage, dependency structures among segments are estimated based on the results of the evolution stage. By separating stream learning into dominant feature extraction and dependency learning, the proposed method is able to analyze streams without prior knowledge.

We apply the proposed method to a hidden dependency learning in a TV drama[1]. In order to obtain flexible segmentation results without being hindered by restricting assumptions (Lane and Brodley, 1999; Poli, 2008), our method does not rely on prior assumptions on a target stream. Instead the prosed method constructs a transitional probability matrix representing dependencies in a stream based on dominant features captured in particles. Because of these novel representation, it is possible to interpret the proposed method as another stream compression method and the proposed method could be applied to applications such as video analysis or stream recommendations. We validate this by regenerating a sequence of anticipated images given a seed image. Numerical experiments indicate that our method provides high detection rates while preserving a good tradeoff between recall and precision.

This chapter is organized as follows. In Section 4.2, we present some related works. Section 4.3 introduces the proposed evolutionary particle filtering and sequential dependency learning scheme. In Section 4.4, we explain the experimental results. Finally, we discuss the characteristics of the proposed method and summarize the proposed work in Section 4.5.

4.2 Related Works

Unsupervised analysis of stream data has been great interest to researchers. Xie et.al represented a layered mixture model for unsupervised stream clustering based on multi-modal feature distributions (Xie et al., 2005). This layered mixture model

[1]We use an American legal drama-comedy, **Boston Legal**

utilizes availability of shots in a news stream which alleviating the difficulty of story boundary finding. Contrary to this, our method is a scheme to divide a TV drama episode into scenes of dominant images where repeating fixed frames do not exist. Gershman introduced a generative model in which a single latent cause is responsible for generating all the observation features (Gershman et al., 2010). We also assume a latent cause generating all frames in a scene but the difference is that particles share features of a dominant image in a collaborative manner. The difficulty is how to construct visual features. Oliva and Torralba introduced a low dimensional representations of the dominant spatial structure of a frame (Oliva and Torralba, 2001). Instead of a set of global features, we build a generational model based on dominant local features. In terms of utilizing bag of visual words, our approach is similar to locally weighted bag of visual words in (Chasanis et al., 2009). However, our method builds a population of particles representing a latent cause of a scene.

If one attempts to build a generative model for given streams, a particle filter method is useful for capturing unobserved changes (Mühlich, 2003). Particle filtering has been widely applied to various tasks such as mobile robot localization (Rekleitis, 2004) or simulating human sentence processing (Levy et al., 2009). One of dominant strengths of particle filtering is its inference capability for latent states (Gershman et al., 2010). Because its simplicity of the *principle of local independence* - if a latent variable underlies a number of observed variables, then conditionalizing on that latent variable will render the observed variables statistically independent (Borsboom et al., 2003), a latent variable model is a promising method for explaining observed variables (Murray and Storkey, 2008; Coquelin et al., 2009). Rather than assuming a prior density functions, we propose an evolution scheme for particle filtering based on (Kwok et al., 2005; Park et al., 2009). In terms of multiple particles, the proposed method is comparable to particle swarm optimization (PSO) (Kennedy and

Eberhart, 1995). However, each particle in our method is not sufficient to represent a solution and our method employs genetic operators to enhance diversity in a population.

Modelling dependencies in sequential data has received strong interest because many real world tasks demand the ability to process patterns in which information content depends not only on static or spatial features but also on temporal order (Barreto and Araújo, 1999). Although there have been attempts to model changing dependency structures (Xuan and Murphy, 2007), a sequential stream with multiple changepoints presents a daunting challenge to unsupervised analysis of the stream. For a stream with unknown number K of partitions, $\pi_1, ..., \pi_k$, such that data is independent across segments then, the given stream observed from time 1 to time T can be represented as Eq. 4.1 and the probability of data from time t to s belong to the same segment is defined as Eq. 4.2 (Fearnhead, 2006).

$$P(y_{1:T}|\pi) = \prod_{k=1}^{K} p(y_{\pi_k}) \tag{4.1}$$

(here, y_{π_k} means frames belonging to kth partition.)

$$\begin{aligned} P(t,s) &= Pr(y_{t:s}|t,s \text{ in the same segment}) \\ &= \int \prod_{i=t}^{s} f(y_i|\theta)\pi(\theta)d\theta \end{aligned} \tag{4.2}$$

(here, θ is a parameter associated with each partition π_k)

The challenge is how to define prior distributions for f and π. It is possible to assign a sophisticated distribution as in (Fearnhead, 2006; Zhu and Song, 2010). However, an assumption on distribution is too rigid for real-world data. Instead of attempting to derive the best regression model for a given stream, we aim to estimate segments in a stream without assuming any prior distribution. An elusive approach would utilize a graph-based method probabilistically merging shots (Sidiropoulos

et al., 2011). Our method also utilizes transitional probability between segments. The difference is that our approach uses transitional probability to estimate temporal order in regenerated images. Several researchers have noted the difficulty due to low-level features (Chasanis et al., 2009; Sidiropoulos et al., 2011). We enhance the robustness by estimating invariant features produced by SIFT (Scalar invariant feature transform) method (Lowe, 1999).

Evolutionary methods in dynamic environments have been actively researched. Wagner et al. (2007) proposed a new "dynamic" genetic programming for dynamic environments. A learning method for temporal sequences requires employing multiple objectives. A data-driven synthesis procedure for real-world object recognition system is introduced in (Krawiec and Bhanu, 2007). Similar to cooperative coevolution in (Krawiec and Bhanu, 2007), our method also utilizes several criteria to impose search bias. In order to consider uncertainties in design variables and problem parameters, a reliability measure is employed in (Deb et al., 2009). Our method enhances the proposed evolution procedure through a modified volatility measure.

4.3 Evolutionary Particle Filtering and Dynamical Sequence Modelling

The core theme of segmentation is very simple. If it would be possible to compute Eq. 4.2 and represent dependencies between parameters associated with each segments, we can describe each segment and relationships between segments. However, it is unrealistic to assume a prior distribution for parameters in Eq. 4.2 if a target

domain is a TV drama where abrupt changes are not unusual.

$$\begin{aligned} P(t,s) &= Pr(y_{t:s}|t,s \text{ in the same segment}) \\ &= \int \prod_{i=t}^{s} f(y_i|\theta)\pi(\theta)d\theta \\ &\approx \frac{1}{N}\sum_{i=1}^{N} f(\mathbf{x}_t^i) \end{aligned} \tag{4.3}$$

(Here, $\mathbf{x}_t^i$ means an individual particle. N is number of selected particles selected from a population of particles, $\mathcal{X}$)

Therefore, we approximate Eq. 4.2 by averaging particles (Eq. 4.3). This approximation has following advantages: 1) major patterns are retained in a population and 2) harmful effect of minor distortions are minimized. In this work, each particle represents a portion of an entire frame and a group of particles represent a frame. In a typical particle filtering, particles converge to an optimal point. This converging deteriorates the representation performance in streams learning. Therefore, evolutionary operators are employed to introduce essential diversity into a population.

Fig. 4.1 shows a basic idea of the proposed method. An initial population ($\mathcal{X}$) is formulated based on a group of randomly extracted SIFT features. Whenever a new frame is observed, likelihood of the new frame given current population is computed. If the computed likelihood is higher than a threshold, evolution continues. If the computed likelihood value is lower than a threshold, it is determined as belonging to a new segment. For a new segment, another population is initialized and evolved. Section 4.3 answer to the question of how to evolve the particles and how to estimate dependencies in a video stream.

4.3.1 Representation

Fig. 4.2 depicts a basic representation scheme. A particle is composed of nodes and each node consists of a SIFT feature and its location in a frame. A set of

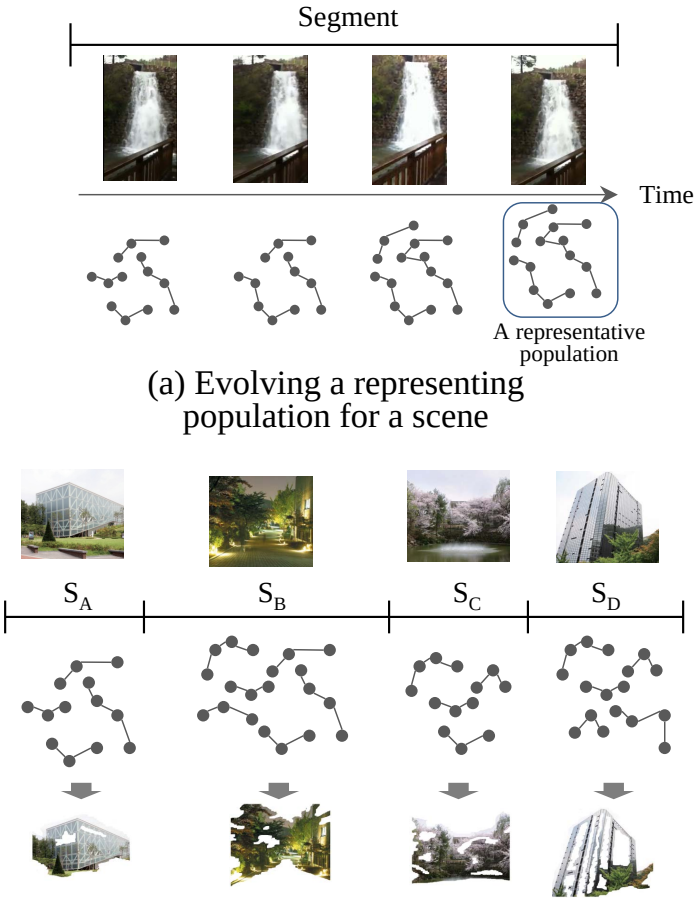
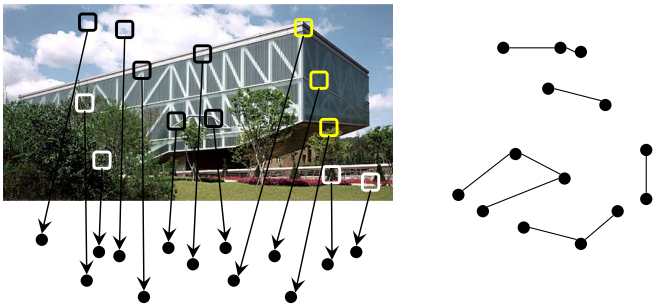
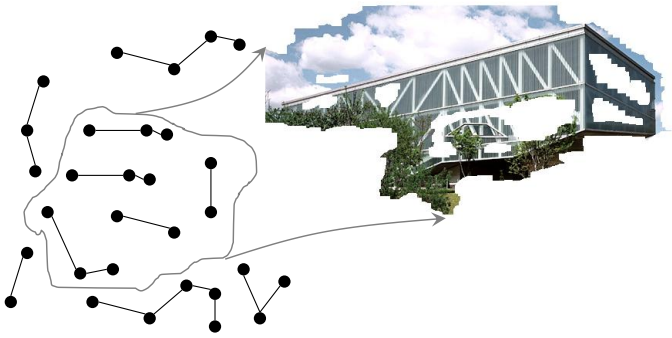


Figure 4.1: Temporal stream compression using the proposed method. (a) A particle population ($\mathcal{X}$) is evolved for a stream interval. (b) A whole stream is represented by a group of particle populations. S_i means the ith segment.



(a) SIFT feature extraction and population formulation



(b) Representation

Figure 4.2: SIFT feature extraction and image representation. (a) SIFT features are extracted from a given frame and an initial population of variable order particles is formed. (b) Particles with higher fitness represent a dominant image in a collaborative manner.

nodes belonging to a particle is selected randomly. Because each node has it own coordinates, it is possible to compare SIFT features allowing slight deviation and reconstruct an image. Our aim is not to converge a population into copies of a dominant particle. Our aim is to find particle representing partial patterns of a dominant image. Therefore, particles with higher fitness represent an image as shown in Fig. 4.2-(b).

4.3.2 Evolutionary Particle Filtering

EM approach

In this section, the proposed scheme is explained in terms of expectation-maximization (EM) framework in Algorithm 3. In a video segmentation, an ideal goal is to calculate the expected coverage-ratio in Eq. 4.4. However, it is intractable to compute Eq. 4.4 if some prior segmentation information is not given. Therefore, we focus on explaining a new frame y_t using a current particle population $\mathcal{X}_t$ in the expectation step (Eq. 4.5).

In the maximization step, a new particle population is generated by Eq. 4.6 using a hybrid method of particle filtering and genetic algorithm, evolutionary particle filtering. In order to enhance degree-of-freedom during evolution, we employ a variable length representation in (Zhang, 2008). An edge is constructed with randomly selected nodes (SIFT features) and this edge is a particle. From two particles participating as operands for crossover $(\mathbf{x}^{i,t-1}_{<1:a>}, \mathbf{x}^{j,t-1}_{<1:b>})$, it is possible that particles with different length is resulted (Eq. 4.9).

Mutation operator in this work is slightly different from a conventional one. Because image regeneration is important subgoal, an existing SIFT feature replaces an

Algorithm 3 Evolutionary PF for segmentation

Input: $D = \{y_{1:T}\}$ (D: SIFT transformed frames)

Output: $\mathbf{X} = \{\mathcal{X}_i : i = 1, ..., n\}$

$\mathcal{X}_i$: a population of particles for ith segment

Begin

1 *Initialization*:

1.1 Draw a set of node (C) based on an initial frame

1.2 Initialize chromosomes of variable orders using C.

2 *Sequence estimation and state summarization*

2.1 Expectation

$$Q(\mathcal{X}_t|\mathcal{X}_{t-1}, y_{s:t}) = E[Pr(y_{s:t}|\mathcal{X}_t)|\mathcal{X}_{t-1}, y_{s:t-1}] \tag{4.4}$$

$$\propto Pr(y_t|\mathcal{X}'_t)Pr(\mathcal{X}'_t|y_{s:t-1}, \mathcal{X}_{t-1}) \tag{4.5}$$

(y_s : the 1st frame of the current segment.)

2.2 Maximization

$$\mathcal{X}_t \leftarrow \arg\max_{\mathcal{X}_t \in \mathbf{X}} Q(\mathcal{X}_t|\mathcal{X}_{t-1}, y_{s:t}) \tag{4.6}$$

2.3 Evolutionary particle filtering

2.3.1 Applying crossover and mutation.

2.3.2 Computing fitness

$$F_c = f_{Cr_{\{a,b,c\}}}(x_t^i) \tag{4.7}$$

(Cr_a: inter-closeness, Cr_b: coverage, Cr_c: penalty)

◦ *Likelihood*

$$F_2 = Pr(y_t|\mathcal{X}_{t-1}), T(y_{s:t}) \tag{4.8}$$

$\begin{cases} F_2 < \gamma\text{: start a new segment and initilize } \mathcal{X}_t \\ F_2 \geq \gamma\text{: } y_t \text{ is regarded as an element of } \mathcal{X}_{t-1} \end{cases}$

element node in a selected particle (Eq. 4.10).

$$\begin{aligned} \mathbf{x}^{i,t}_{<1:a'>} &= \mathbf{x}^{i,t-1}_{<1:k_1>} \oplus \mathbf{x}^{j,t-1}_{<k_2+1:b>} \\ \mathbf{x}^{j,t}_{<1:b'>} &= \mathbf{x}^{i,t-1}_{<1:k_2>} \oplus \mathbf{x}^{j,t-1}_{<k_1+1:a>} \end{aligned} \tag{4.9}$$

(here, $\oplus$ means a concatenation operator)

$$\mathbf{x}^{i,t}_{<1:a>} \leftarrow \mathbf{x}^{i,t-1}_{<1:k-1>} \oplus x'_k \oplus \mathbf{x}^{i,t-1}_{<k+1:a>} \tag{4.10}$$

In the proposed method, the role of mutation is prevent sample impoverishment and maintain diversity in a population. Therefore, we use higher crossover and mutation parameter during evolution (Table 4.1) and guaranteed minimum evolution time of $0.1s$ for each segment.

Table 4.1: Evolution parameters

	Crossover	Mutation
Value	0.5	0.05

Eq. 4.7 explains fitness function for a particle $\mathbf{x}^i_t$. Eq. 4.7 reflects closeness (Cr_a) of SIFT words in a particle and the coverage ratio (Cr_b) by a particle. Cr_a evaluates distance between nodes in a particle using Eq. 4.11.

$$Cr_a = \sum_{j=1}^{J} |S_j - Cent|^2 \tag{4.11}$$

(S_j: jth node in a particle, Cent: a centroid coordinates of all nodes in a particle)

If distance among nodes in a particle is too long, it is likely that the particle is not robust enough because some part of a particle would disappear at the next frame. Therefore, Cr_a prefers lower average distance. Cr_b measures representing capability of a particle. If a particle has too many nodes, it is possible that this particle has too detailed information. In order to obtain more abstract level of representation,

we prefer shorter particles. Thus, we assign penalty proportional to its length (Cr_c). Relative contribution of $Cr_a,\ Cr_b,\ Cr_c$ is determined empirically.

Likelihood of a new frame is generated by the current particle population ($\mathbf{X}$) is computed by Eq. 4.8. Ideally, the likelihood of a new frame y_{t+1} should be estimated using Eq. 4.12.

$$p(y_{t+1}|y_{s:t}, m) = \int [\prod_{i=s}^{t+1} p(y_i|y_{s:i-1}, \theta, m)] p(\theta|m) p(m) d\theta \tag{4.12}$$

(here, θ means hyper-parameters associated with each partition). In the proposed method, we do not assume any specific distribution for each term in Eq. 4.12. Therefore, cover-ratio of current $\mathcal{X}_t$ by Eq. 4.8 substitutes likelihood by Eq. 4.12. In order to alleviate volatility during early evolution phase, we consider volatility measured by Algorithm 5.

4.3.3 Sequential Dependency Learning and Volatility Measure

Because Algorithm 3 produces only the estimated scene change-points, the resulted segments set is not suitable for sequential dependency learning. Thus we need a method to make clusters of similar scenes. Algorithm 4 performs this task and compute transitional probability between clustered scenes.

Input to Algorithm 4 is $\mathbf{X} = \{\mathcal{X}_i : \ i = 1, \cdots, n\}$ obtained from Algorithm 3. Output of Algorithm 4 is a transitional probability matrix of estimated segments. The goal is to 1) reduce the size of $\mathbf{X}$ by removing similar $\mathcal{X}_i$ in $\mathbf{X}$ and to 2) compute the transition probability between elements in the resulted $\mathbf{T}$. A loop is executed until each $\mathcal{X}_i$ is compared to each other. If $\mathbf{T}$ is empty, a $\mathcal{X}_1$ is stored in $\mathbf{T}$ as τ_k ($k = 1$). Then, $\mathcal{X}_j$ ($j < 1$) is compared to τ_k and if $\tau_k = \mathcal{X}_j$ with slight deviation ϵ, $\mathcal{X}_j$ is ignored. If $\tau_k \neq \mathcal{X}_j$ and $\mathcal{X}_j$ is not in $\mathbf{T}$, then $\mathcal{X}_j$ is added to $\mathbf{T}$ as τ_{k+1}. In order to obtain a transitional probability matrix $\mathbf{M}$, $\mathbf{X}$ is converted into τ notation

Algorithm 4 Sequential Dependency Learning

Input : $\mathbf{X} = \{\mathcal{X}_i | i = 1, \cdots, n\}$

Output : $\mathbf{T} =< \tau_1, \cdots, \tau_k >$ and $\mathbf{M}$

($\mathbf{T}$: a reduced segment set based on $\mathbf{X}$, $\mathbf{M}$: a transitional probability matrix for $\mathbf{T}$)

Initialize $\mathbf{T}$ and $k = 1$

if $|\mathbf{T}| = 0$, add $\mathcal{X}_1$ to $\mathbf{T}$ as τ_k and $j = 2$

else while $j \neq n$

if $g(\mathcal{X}_j, \tau_k) = 1$ & $\mathcal{X}_j \notin \mathbf{T}$, then

$k = k + 1$, add $\mathcal{X}_j$ to $\mathbf{T}$ as τ_k and $j = j + 1$

else $g(Pf_j, \tau_k) = 0,\ j = j + 1$

here,

$$g(\mathcal{X}_j, \tau_k) = \begin{cases} 1, \text{ if } \mathcal{X}_j \neq \tau_k \\ 0, \text{ if } \mathcal{X}_j = \tau_k \text{ allowing some deviation } \varepsilon \end{cases} \tag{4.13}$$

$\circ$ Computing Transitional Probability $\mathbf{M}$ ($|T| \times |T|$)

For $i = 1, ..., n$ and $k = 1, ..., |T|$

if $g(\mathcal{X}_i, \tau_k) = 0$, represent $\mathcal{X}_i$ as τ_k ($\tau\mathbf{X}$)

Each element m_{ij} of $\mathbf{M}$ is defined as

$$m_{ij} = \frac{\#\tau_i \rightarrow \tau_j \text{ transtions}}{\#\tau_i \text{ transtions}} \tag{4.14}$$

here,

$-$"$\#\tau_i \rightarrow \tau_j$ transtions" is # of transitions between ith element and i+1th element in $\tau\mathbf{X}$ if ith element $= \tau_i$ and $i + 1$th element $= \tau_j$

$-\#\tau_i$ transtions is # of transitions between ith and $i + 1$th element in $\tau\mathbf{X}$ when ith element is τ_i

by comparing each element in $\mathbf{X}$ and $\mathbf{T}$ ($\tau\mathbf{X}$). In order to obtain a transitional matrix $\mathbf{T}$ of $|T| \times |T|$, every transition between ith element and $i+1$th element in $\tau\mathbf{X}$ is considered. An element of $\mathbf{M}$ m_{ij} is defined as a fraction of the number of transitions $\tau_i \rightarrow \tau_j$ to the number of transition started with τ_i (Eq. 4.14).

With evolutionary approach, there exists a danger of false estimation due to pre-evolved population. In order to prevent such danger, we introduce a volatility measure to estimate convergence tendency in a population. There has been huge interest on volatility measure in the investment community (Ambrosio and Jr., 2008). We employ a modified average true range (ATR). A modified average true range is adopted to measure stability of a population and computes gradient by comparing the best fitness of previous generation to the best and the worst fitness of current generation (Algorithm 5). The measured volatility modifies the threshold for likelihood computation (Eq. 4.17).

4.3.4 Image Regeneration

As in Fig. 4.1, each scene has an associated particle population. These particles retain dominant image features for a corresponding segment. An image is regenerated using these particles. In the proposed method, each node in a particle has location information (central point of each image patch) for a corresponding image. An image is regenerated by combining image patches in particles based on the location information. There exists a clear benefit in the image regeneration. With regenerated images, it is possible to verify the results in a visual way.

4.4 Experimental Results

We report performance of the proposed method by comparing to human-evaluated ground truth. We analyze performance in terms of accuracy, fitness, and image

Algorithm 5 Volatility Measure

Input : $D = \{y_{s:t-1}|\text{stream data from s to t-1}\},\ Pf_j$

Output : α (volatility measure)

While termination criteria

For each generation t

$$\begin{cases} g_{1,t} &= H_t - L_t \\ g_{2,t} &= H_t - H_{t-1} \\ g_{3,t} &= H_{t-1} - L_t \end{cases} \tag{4.15}$$

(H_t: the best fitness at the current generation, L_t: the worst fitness at the current generation)

$$\alpha_t = g_{1,t}^2 + g_{2,t}^2 + g_{3,t}^2 \tag{4.16}$$

γ' in Eq. 4.8

$$\gamma' = \gamma \times \frac{1}{1 + e^{-\alpha_t}} \tag{4.17}$$

regeneration. Prior to provide experimental results, we report estimation by human participants in order to explain difficulty of the task.

4.4.1 Data and Human Evaluations

We asked 10 students (10 undergraduates) to evaluate 3 episodes in terms of dominant image changes. The length of episodes are 42 minutes, 41 minutes 40 seconds, and 41 minutes 26 seconds. Fig. 4.3 presents more detailed information on the hu-

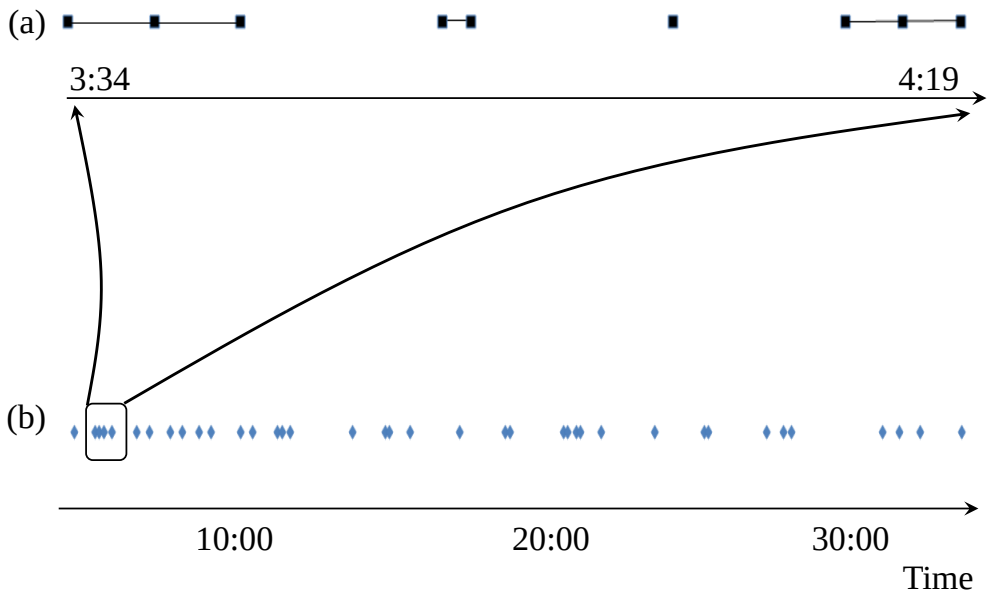


Figure 4.3: A detailed human estimation result. Fig. 4.3-(a) is an interval description for averaged changepoints. An averaged changepoint is derived from a interval. If an estimation results belong to a interval, then it is regarded as an accurate estimation.

man estimations. As in Fig. 4.4, human evaluators did not agree on one changepoint. Therefore, we constructed an interval based on the human evaluations. If an estimation point provided by the proposed method belongs to this interval, the estimation point is regarded as a correct one. Table 4.2 provides statistical summarizations of these intervals. A minimum length of 1 second is reasonable, but a maximum length of 9 second seems to be too long. These long intervals belong to some scenes where

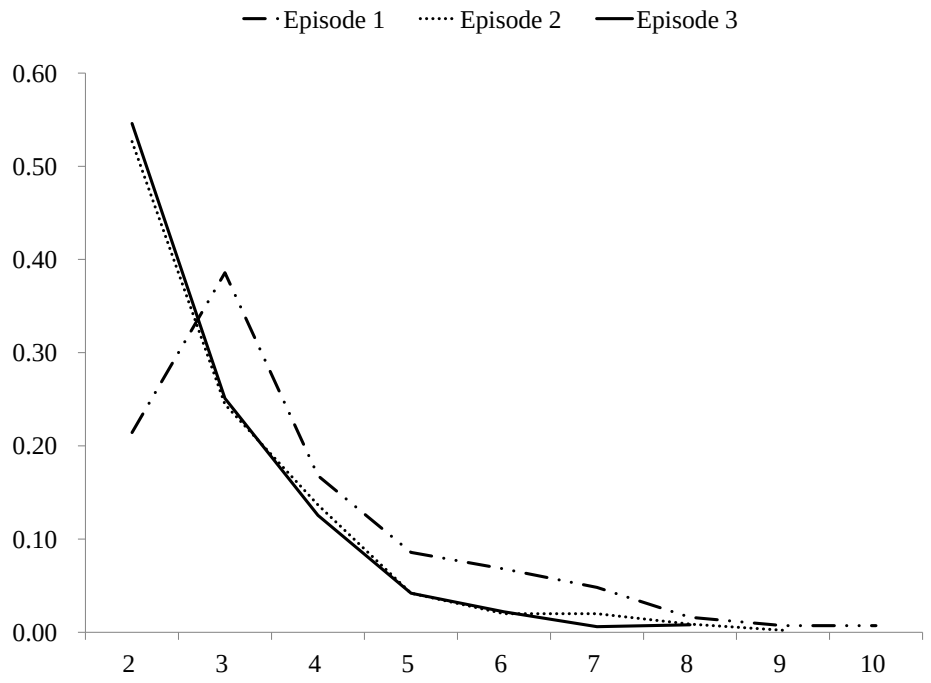


Figure 4.4: Distributions of human evaluations. It could be assumed that a scene change would be estimated unanimously, Fig. 4.4 shows that this expectation does not hold.

varying landscape of the background city - Boston - is being shown rapidly.

4.4.2 Segmentation and Dependency Learning

Estimation Results

A first series of experiments was carried out with the proposed method on episodes of a TV series and their results were compared to the human evaluations. The

Table 4.2: Statistical summarization

Episode I	Average	Std.	Min	Max
Episode 1	1.84s	1.03s	1.00s	8.00s
Episode 2	2.10s	1.15s	1.00s	7.00s
Episode 3	1.88s	1.03s	1.00s	9.00s
"Std." means "standard deviation"				

estimation results are shown in Table 4.3 and Table 4.4. In order to demonstrate the performance in a more traditional way, we provide recall and precision.

$$\text{Precision} = \frac{\#(\text{Correctly estimated changepoints})}{\#(\text{Total estiamted changepoints})}$$

$$\text{Recall} = \frac{\#(\text{Correctly estimated changepoints})}{\#(\text{Total ground truth intervals})}$$

$$F_1 = \frac{2 \times Recall \times Precision}{Recall + Precision}$$

Table 4.3: Proposed method and human evaluations

		Episode 1	Episode 2	Episode 3
Human	No. of changes	981	798	818
	No. of intervals	560	444	498
Computer	No. of changs	2206	1746	1576

Table 4.3 reports initial estimation results and Table 4.4 shows the precision and recall performance. In order to validate performance of the proposed method, we performed a comparison experiment based on the color histogram method. In the

Table 4.4: Precision and Recall Performance

	Episode 1	Episode 2	Episode 3
Precision	0.273	0.410	0.461
Recall	0.614	0.896	0.889
F_1	0.378	0.562	0.607

case of detecting changes in dominant images, one would assume that changes in color could be a good clue.

Particle Population

With Fig. 4.5 and Table 4.5, we analyze the changes in a population during evolution. We depict a fitness curve of the best particle during an estimated segment in Fig. 4.5. Because particles with higher fitness are retained at each generation, the proposed method was able to achieve the required stability in relatively short generations. Table 4.5 reports trends in ratio of each order in a population. In Table 4.5, 1/4, 2/4, 3/4, and, 4/4 means 1st quantile, 2nd quantile, 3rd quantile, and, 4th quantile in a measured population.

We observe ratio of each sub-group in a population based on its order. Evolution starts with initial order of 5. Table 4.5 shows that particles with middle order (order 7, 8, 9, 10) increase their proportion. This is the effect of the fitness function (Eq. 4.7). Particles with shorter order are disfavored due to lack of representational capability. Particles with longer order are penalized due to their longer length. Through collaborations among particles of limited representation capability, the proposed method generated discernible images in Fig. 4.6 and Fig. 4.7.

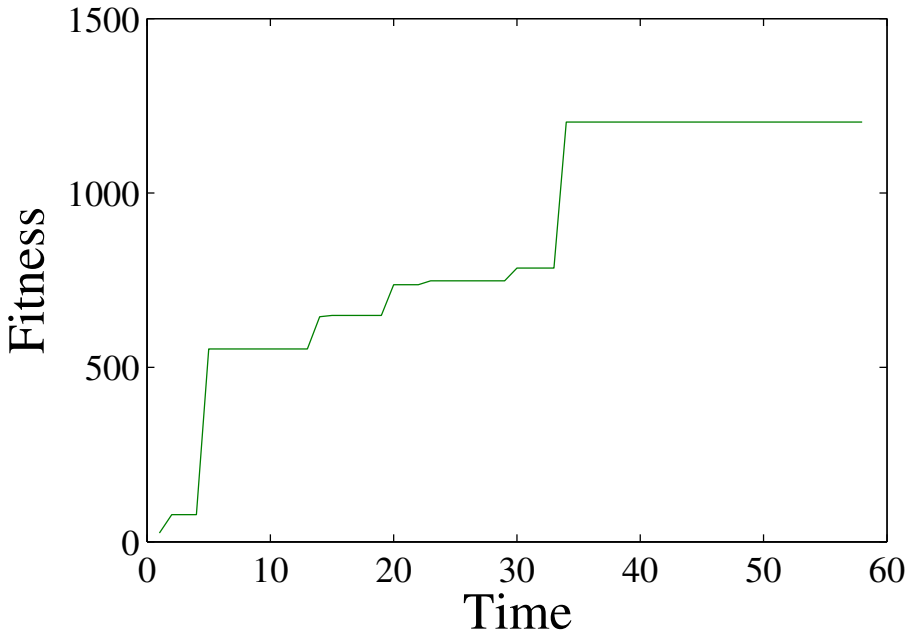


Figure 4.5: Fitness curve in a segment

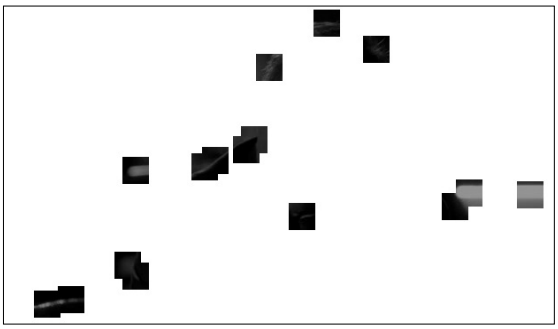
In order to present the effect of order of a particle, we compare images generated with different orders. Fig. 4.6-(a) was generated by particles of order 1. Fig. 4.6-(b) was generated by the proposed method. Fig. 4.6-(c) was generated by a particle of order 600. Actually, Fig. 4.6-(a) corresponds to a process of feature selection. Fig. 4.6-(c) is an attempt to represent an image with a particle. The proposed method generated the most discernible image. Therefore, the propose method is more suitable for segmentation and compression (summarization) of a given video stream.

Table 4.5: Trends in order distribution during evolution

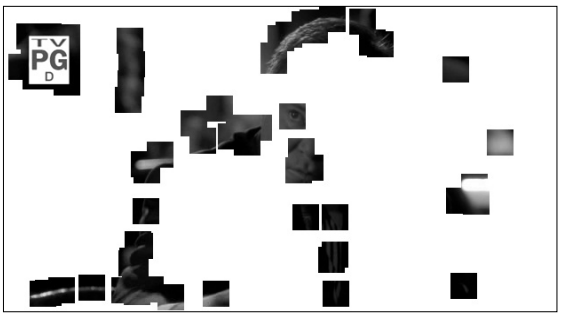
Order	Initial ratio	1/4	2/4	3/4	4/4
5	1.00	0.16	0.04	0.04	0.04
6	0.00	0.32	0.16	0.12	0.06
7	0.00	0.28	0.24	0.20	0.20
8	0.00	0.16	0.20	0.18	0.14
9	0.00	0.06	0.15	0.18	0.16
10	0.00	0.02	0.13	0.12	0.16
11	0.00	0.00	0.08	0.10	0.08
12	0.00	0.00	0.00	0.04	0.12
13	0.00	0.00	0.00	0.00	0.00
14	0.00	0.00	0.00	0.00	0.02
15	0.00	0.00	0.00	0.02	0.02

Dependency Learning

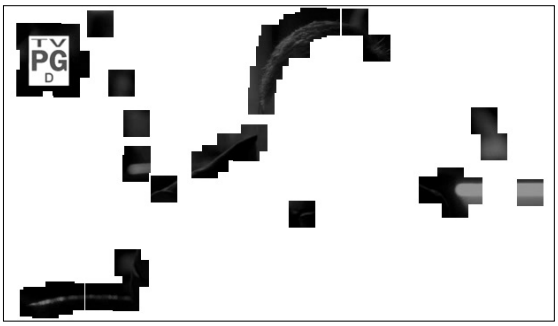
In order to validate performance of dependency learning, we regenerated anticipated sequence of scenes for a given seed image. In detail, we generate an image of a succeeding segment for the given seed image based on the transition probability matrix obtained by algorithm 2. We estimated another succeeding segment using the regenerated image as a seed. We repeated this process for three times. The result is shown in Fig. 4.7 and Table 4.6. The black and white images in the right column of Fig. 4.7 represent dominant images of estimated scenes. The color images in the left column of Fig. 4.7 are real dominant images of the original scenes. Except the second image, the remaining three images are identical to the original ones. The



(a) Particles of order 1



(b) Particles of mixed orders (the proposed method)



(c) A particle of big order

Figure 4.6: Effect of order on an image generation

second images in the original sequence and the anticipated sequence also share a common character (a bald man). Therefore, the proposed method supposed a near perfect sequence for the given seed image. In order to verity the performance, we

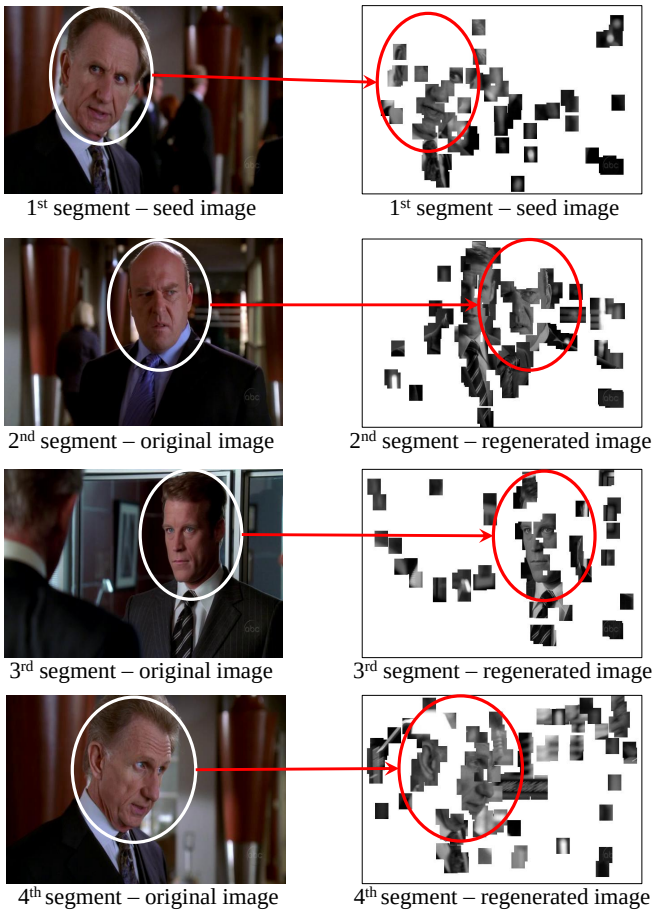


Figure 4.7: Regenerated images based on the estimated transitional probability matrix. By comparing images in circles, we are able to determine whether dominant features were regenerated.

repeated this experiment for 10 seed images, and the ratio of correctly estimated sequence is reported in Table 4.6.

Table 4.6: Sequence regeneration performance

# Segment	Accuracy
2	43.4%
3	15.0%
4	10.0%

4.4.3 Comparison with Other Method

Intuitively, a color profile of a temporal stream is a reasonable cue for detecting changes. In order to verify the performance, we compare the proposed method with segmentation based on a color histogram. Table 4.7 reports performance based on

Table 4.7: Performance based on the color histogram method

	Case 1	Case 2	Case 3	Case 4	Case 5
No. of Changes	48	95	192	406	1008
Precision	0.69	0.71	0.59	0.46	0.49
Recall	0.040	0.081	0.14	0.23	0.60
F_1	0.076	0.15	0.22	0.31	0.54

the color histogram method in terms of precision, recall, and F_1 score for the 3rd episode. We observed trends according to threshold for determining change. When one compared F_1 scores, the superiority of the proposed method is clear. For all the thresholds, the color histogram scored low F_1 values against 0.607 of the proposed method.

4.5 Discussion and Summarization

In this chapter, we have introduced a temporal stream learning scheme with two components: evolutionary particle filtering for segmentation and sequential dependency learning. We proposed an evolutionary particle filtering based model to utilize dominant images in a frame with slight distortion. With sequential dependency learning module, we have demonstrated that it is possible to interpret the proposed method as a method for temporal steam compression. The proposed method is a first step towards more ambitious goal of real-world video stream analysis and recommendation. Because we did not employ sophisticated prior assumptions for estimation, the proposed method possesses more flexibility for representing real-world video streams.

There are several possible directions for future works. Although the proposed method is promising one, it may be hindered by search biases of the evolutionary approach. Under restriction of real-time processing of a given video stream, one should evolve a population in limited generations. Therefore, a further research is required to restrict a possible search space. In order to recommend a video stream based on the proposed method, we need a method to combine several segments and compare the combined segments with other video streams.

Chapter 5

Multiple Stream Learning

We demonstrate how to perform semantic segmentation in multichannel streams where the number of changepoints is unknown. In contrast to previous temporal segmentation approaches that employ mostly pre-defined assumptions or features in a single channel, we introduce a technique that jointly exploits temporal patterns in an image channel and a sound channel. This approach is based on the use of latent variables, and is related to work on non-parametric Bayesian estimation and Gaussian mixture models. For an image channel, the estimation method is built upon the well-known method of hierarchical Dirichelt process (HDP), first by estimating a HDP model benefiting from inherent hierarchical structures, and then sequentially building another HDP model based on the likelihood. For a sound channel, segmentation is performed by constructing a dialogue model employing Gaussian mixture models. Instead of assuming a state transition probability, the proposed method builds a dialogue interval from the recognized speakers. A semantic segment is approximated by merging two channels dynamically. Our approach can be useful, for example in a stream recommendation domain, even without prior knowledge. The proposed method is evaluated on three episodes of a TV drama and numerical re-

sults indicate that our method provides high detection rates while preserving a good trade-off between recall and precision.

5.1 Multichannel based Approach

In this chapter, we introduce a multichannel based sematic segmentation method. This approach is based on the use of latent variables, and is related to work on non-parametric Bayesian estimation and Gaussian mixture models. We divide a stream into an image channel and sound channel. For an image channel, we propose sequential hierarchical Dirichlet process (sHDP), capable of utilizing a hierarchical structure inherent in dynamic sequential multidimensional data such as TV drama. For a sound channel, we adopt a speaker recognizing model using MFCC (Mel Frequency Cepstral Coefficient). A dynamic channel merging scheme combines estimation by each channel seamlessly. A TV drama episode has various characteristics. There is temporal consecutiveness one could exploit and conceptual hierarchies composed of unlimited number of image features, words, objects, sentences, and stories (Mittal, 2006). In a TV drama episode, there also exist multiple alternating stories sharing common characters and backgrounds. In this approach, we adopt a latent variable model in order to explain frames and dialogues belonging to a story segment.

Our method achieves promising performance by approximating a distribution explaining a number of consecutive frames and dialogue intervals. Various researchers have reported sequentiality analysis methods based on a sophisticated prior distribution. However, it is unreasonable to assume that story segments could be determined by a prior distribution. Therefore, we estimate semantic changes by considering likelihood for changes in both of an image channel and a sound channel. Because plenty of speaker recognition techniques have been introduced, we estimate intervals in a sound channel based on recognition of speakers.

We apply the proposed method on distinguishing story changes in an episode of a TV drama. Estimating changepoints in a TV drama is often a first step towards the more ambitious goal of contents recommendation; a latent model for a semantic segment could lead to a semantic descriptor explaining a segment (scene) in terms of modality distributions. The present paper emphasizes the change detection problem focused on multiple channels in a stream. We report the performance of the proposed method by comparing to the human estimated ground-truth for 3 episodes of a target TV drama.

This paper is organized as follows. In Section 5.2, we review the previous HDP models. Section 5.3 introduces the proposed semantic segmentation scheme. In Section 5.4, we explain the representation and experimental results. Finally, we discuss the characteristics of the proposed method and summarize this work in Section 5.5.

5.2 Related Works

5.2.1 Approaches for Temporal Stream Analysis

It is important to take account the dynamic changes over time because dynamic change is an inherent feature of many real-world data sets (Wang and McCallum, 2006). A Topics over Time (TOT) model introduced a mechanism for capturing how the structure changes over time and X. Wei et al. presented a dynamic mixture model takes into consideration the temporal information implied in the data (Wang and McCallum, 2006; Wei et al., 2007). However, previous approaches utilized very limited characteristics such as pre-defined short cuts and repetition of frames for analyzing multimodal streams (Chaisorn et al., 2003; Poli, 2008; Manson and Berrani, 2010; Ibrahim et al., 2010). Contrary to these approaches, a dynamic statistical

model is capable of analyzing multimodal streams based on the more inherent characteristics such as likelihood of state change and a mixture of hidden variables.

5.2.2 Hierarchical Dirichlet Process

There exist various hierarchical approaches suitable for a multimodal stream model. A hierarchical approach, the hierarchical Dirichlet process (HDP) (Ghahramani, 2005; Teh et al., 2005), has been proposed to the task of model-based clustering of grouped data through sharing the same set of discrete parameters. It has been shown that HDP has potential as a building block in models for time-evolving data (Teh et al., 2005; Teh and Jordan, 2010; Ren et al., 2008). Teh et al. (Teh et al., 2005) presented a nonparametric Bayesian approach (HDP-HMM) for temporal sequence learning by replacing the set of conditional finite mixture models underlying the classical HMM with a hierarchical Dirichlet process mixture model. Recently, sticky HDP-HMM (Fox et al., 2008), dynamic HDP (Ren et al., 2008), and time-varying DPM (Caron et al., 2007) introduced promising approaches for temporal sequence learning. The sticky HDP-HMM employs an alternative sample transition distribution in order to address the problem of state sequence with unrealistically fast dynamics having large posterior probability. Ren et al. (Ren et al., 2008) introduces a dynamic hierarchical Dirichlet process (dHDP) to incorporate time dependence. TVDPM (Caron et al., 2007) constructs a model by random deleting and sampling of a new allocation variables. Ahmed and Xing (Ahmed and Xing, 2010) introduced a model, iDTM (infinite Dynamic Topic Models) that can adapt the number of topics, the word distributions of topics, and the trend over time. P. Orbanz et al. (Orbanz et al., 2007) introduced adaptive clustering method for video segmentation.

A hierarchical Dirichlet process (HDP) is a nonparametric approach to the modelling of groups of data, where each group is characterized by a mixture model and

mixture group to be shared between groups. The Dirichlet process is a measure on measures (Ferguson, 1973). A hierarchical Dirichlet process is a distribution over a measurable space Θ. The HDP model is represented as:

$$G_0|\gamma, H \sim DP(\gamma, H)$$

$$G_j|\alpha_0, G_0 \sim DP(\alpha_0, G_0)$$

This process defines a set of random probability measure G_j, one for each group, and a global random probability measure G_0. The random measure G_j are conditionally independent given G_0, with distributions given by a Dirichlet process with base probability measure G_0. The baseline probability measure H provides the prior distribution for the factor θ_{ji}. Each θ_{ji} is a factor corresponding to a single observation x_{ji} whose likelihood is given by $x_{ji}|\theta_{ji} \sim F(\theta_{ji})$. The distribution G_0 varies around the prior H, with the amount of variability governed by γ. The actual distribution G_0 over the factors in the j^{th} group deviates from G_0, with the amount of variability governed by α_0 (γ, α_0: concentration parameters).

Because there exist undefined number of objects in an episode of a TV drama, a data-driven non-parametric approach is suitable for analyzing TV dramas. In order to explain the existence of inherent hierarchies in a video stream, we utilize the property of Chinese restaurant metaphor (Teh et al., 2005) in cluster ensemble building (Ahmed and Xing, 2008; Wang et al., 2010). Due to its tendency that a popular restaurant remains popular, Chinese restaurant franchise is a good candidate to describe a set of data belonging to a same cluster.

One serious limitation of the standard HDP is that it inadequately models the temporal persistence of states (Fox et al., 2008). Ren et al. (Ren et al., 2008) and Fox et al. (Fox et al., 2008) proposed HDP-HMM model and dHDP model in order to improve HDP's performance on sequential data.

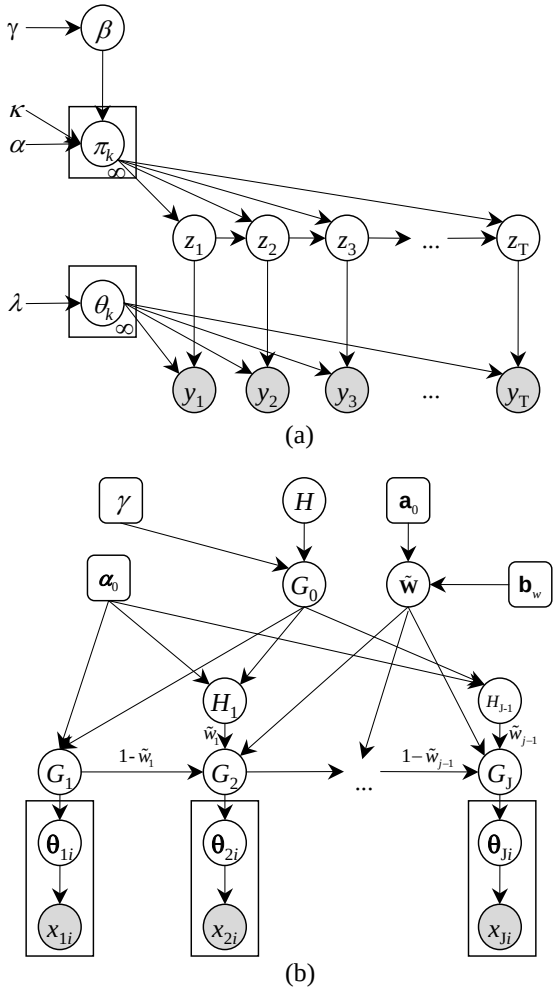


Figure 5.1: (a) Graph of the sticky HDP-HMM. $\beta \sim GEM(\gamma)$ and observations are generated as $y_t \sim F(\theta_{z_t})$. y_i is an observed data and κ values increase the prior probability $E[\pi_{jj}]$ of self-transitions. $\theta_k \sim \mathrm{H}(\lambda)$ (image derived from (Fox et al., 2008)). (b) General graphical model for dynamic HDP (image derived from (Ren et al., 2008)).

5.2.3 Sticky HDP-HMM and Dynamic HDP

The hierarchical Dirichlet process hidden Markov model (HDP-HMM) was introduced to allow more robust learning of smoothly varying dynamics (Fig. 5.1 (a)). One serious limitation of the standard HDP-HMM is that it inadequately models the temporal persistence of states. When modelling systems with state persistence, HDP-HMM prior allows for state sequences with unrealistically fast dynamics to have large posterior probability (Fox et al., 2008). In addition, this problem is compounded by the tendency of the Chinese restaurant franchise generating some kind of loyalty. To address these issues, another method, the sticky HDP-HMM, to sample transition distributions π_j is proposed as follows:

$$\pi_j \sim DP(\alpha + \kappa, \frac{\alpha\beta + \kappa\delta_j}{\alpha + \kappa}) \tag{5.1}$$

Here, π_j is a state-specific transition distribution. z_t denotes the state of the Markov chain at time t and $z_{t+1} \sim \pi_{z_t}$. $(\alpha\beta + \kappa\delta_j)$ indicates that an amount $\kappa > 0$ is added to j^{th} component of $\alpha\beta$.

The dynamic hierarchical Dirichlet process (dHDP) was developed to model the time-varying statistical properties of sequential data (Ren et al., 2008; Dunson, 2006). dHDP extends HDP to incorporate time dependence and considers the following important features: (i) two data samples drawn at proximate times is likely to share the same underlying model parameters; and (ii) it is possible that temporally distant data samples may also share model parameters. Based on these assumptions, a distribution G_{j-1} is likely related to G_j.

Dunson (Dunson, 2006) proposed a model in which G_j and G_{j-1} shares some features but some innovations may also occur. This structure is represented in dHDP as follows:

$$G_j = (1 - \widetilde{w}_{j-1})G_{j-1} + \widetilde{w}_{j-1}H_{j-1} \tag{5.2}$$

H_{j-1} is an "innovation distribution" drawn from $DP(\alpha_0, G_0)$. and $\widetilde{w}_{j-1} \sim Be(a_{w(j-1)}, b_{w(j-1)})$. When the discrete base distribution drawn from $DP(\gamma, H)$ then

$$H_{j-1} = \sum_{k=1}^{\infty} \pi_{J,k} \delta_{\theta_k{}^*} \tag{5.3}$$

where $\{\theta_k^*\}$ are the global parameter components drawn independently from the base distribution H and the different weights $\pi_j|\alpha_{0,j}, \beta \sim DP(\alpha_{0,j}, \beta)$.

5.2.4 Speaker Recognition

The speaker recognition algorithms consist of training sessions and operating sessions (Muda et al., 2010). For training and operating, the extraction of the best parametric representation of acoustic signals is important. A feature extraction method, MFCC is based on human hearing perceptions which cannot perceive frequencies over 1Khz. MFCC has two types of filter which are spaced linearly at low frequency below 1000 Hz and logarithmic spacing above 1000Hz. MFCC is composed of seven computational steps: pre-emphasis (the passing of signal through a filter), framing (segmenting the speech samples into a small frame), hamming windowing (integrating all the closest frequency lines), fast fourier transform (converting samples from time domain into frequency domain), Mel filter bank processing (computing a weighted sum of filter spectral components), discrete cosine transform (converting the Mel spectrum into time domain), and delta energy and delta spectrum (adding features related to the change in cepstral features).

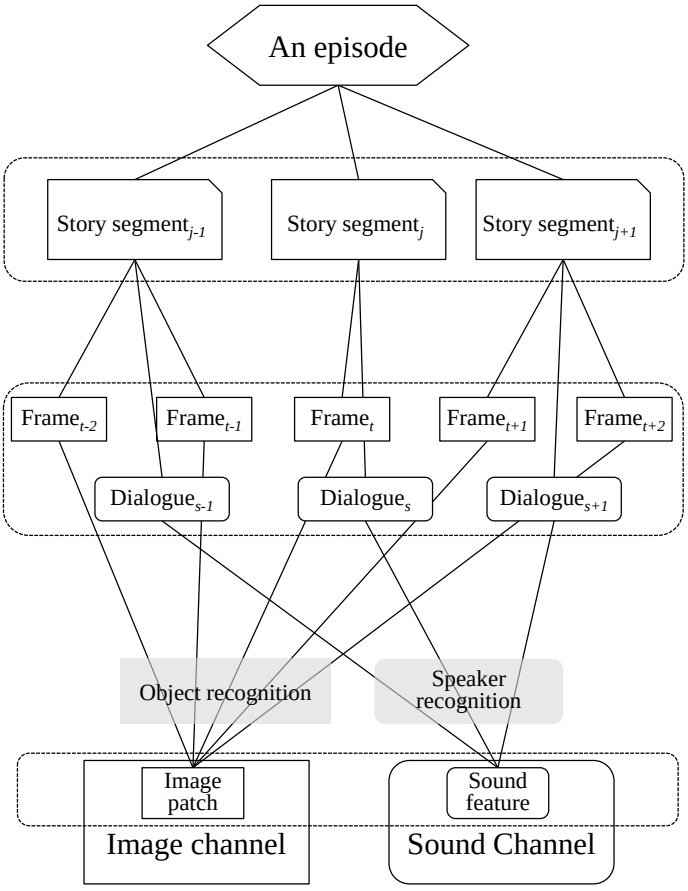


Figure 5.2: A hierarchical organization of data; it is possible to interpret an episode in a hierarchy. Basic elements are image patches and sound features. Image features construct a frame and a dialogue is composed of sound features. A story segment consists of a set of frames and dialogues. When a set of image patches is converted into a frame, an object would be recognized. Similarity, a speaker recognizer would help a conversion from sound features into a dialogue.

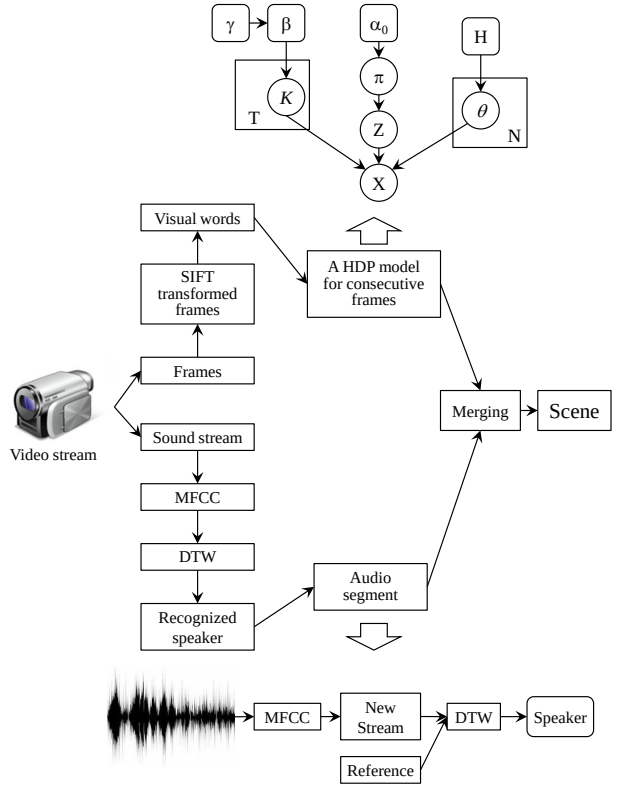


Figure 5.3: Segmenting schematic: a video stream is divided into an image channel and a sound channel. In this schematic, MFCC means mel frequency cepstral coefficients.

5.3 Semantic Segmentation Scheme

When a multidimensional sequence is given, it is difficult to find relations in the given sequence using conventional HMM learning methods in terms of accuracy or computational cost (Wang et al., 2006). An efficient Bayesian changepoint detection

method was introduced in (Xuan and Murphy, 2007). Our method takes a similar approach to detect segment changepoints, but we introduce a composite method for mulchannel streams.

As shown in Fig. 5.2, a typical stream data is composed of an image channel and a sound channel. In addition, it is possible to assign a hierarchical structure for each channel. In order to utilize the existence of multiple channels and hierarchical structure, we propose divide and conquer approach as in Fig. 5.3. For an image channel we propose sequential HDP (sHDP) utilizing likelihood of data generated by a latent model. The utilization of likelihood makes the proposed method become more robust to abrupt changes because the effect of the difference in consecutive scenes is diminished. For a sound channel we develop a model based on speaker recognition using MFCC.

5.3.1 Sequential HDP

We explain the segment changes in an image channel by employing the Chinese restaurant franchise (CRF) analogy. In the CRF, the metaphor of the Chinese restaurant process is extended to allow multiple restaurants which share a set of dishes. The first customer at each table of each restaurant orders one dish and it is shared among all customers who sit at that table. Multiple tables in multiple restaurants can serve the same dish (Teh et al., 2005).

In the case of TV dramas, an image channel in an episode is explained as a set of story segments where visual word[1], θ_{ji}, composing the stream are customers in the CRF analogy. Each restaurant corresponds to a story in Fig. 5.2. These video stream share a global set of conceptual objects (menu) ϕ_k. Conceptual objects mean a set of image objects and this setting can represent the sharing of common characters

[1]In this implementation, image patches is composed of a set of visual words.

Algorithm 6 Multichannel modelling

Input: video stream

Output: A set of segmented scenes S

Begin

∘ *For a SIFT transformed image channel*

$$G_0|\gamma, H \sim DP(\gamma, H)$$

I-1. Sampling a group of θ (visual word) (Eq. 5.8)

I-2. Assigning identity (Eq. 5.7)

I-3. For a new SIFT transformed frame

- **Prediction** Calculate likelihood L

$$L(x_{ji}|G_j) = \begin{cases} L < \tau_I, \text{ Candidate of a new scene} \\ \qquad G_{j+1} \\ L > \tau_I, \ G_j \text{ enhancement} \rightarrow \text{Go to I-1} \end{cases}$$

∘ *For a sound channel*

S-1. MFCC Processing.

S-2. Transformation into low dimensional sub-channel.

- **Prediction** Calculate MFCC coefficient (r)

$$\begin{cases} if \ r > \tau_S \text{ Candidate of a new segment} \\ else \ r < \tau_S \qquad \text{Go to step S-1} \end{cases}$$

∘ *Dynamic channel merging*

$F(x_t, s_j|G_L) = f(\triangle L_t, I(S_t))$ (refer to section 5.3.4)

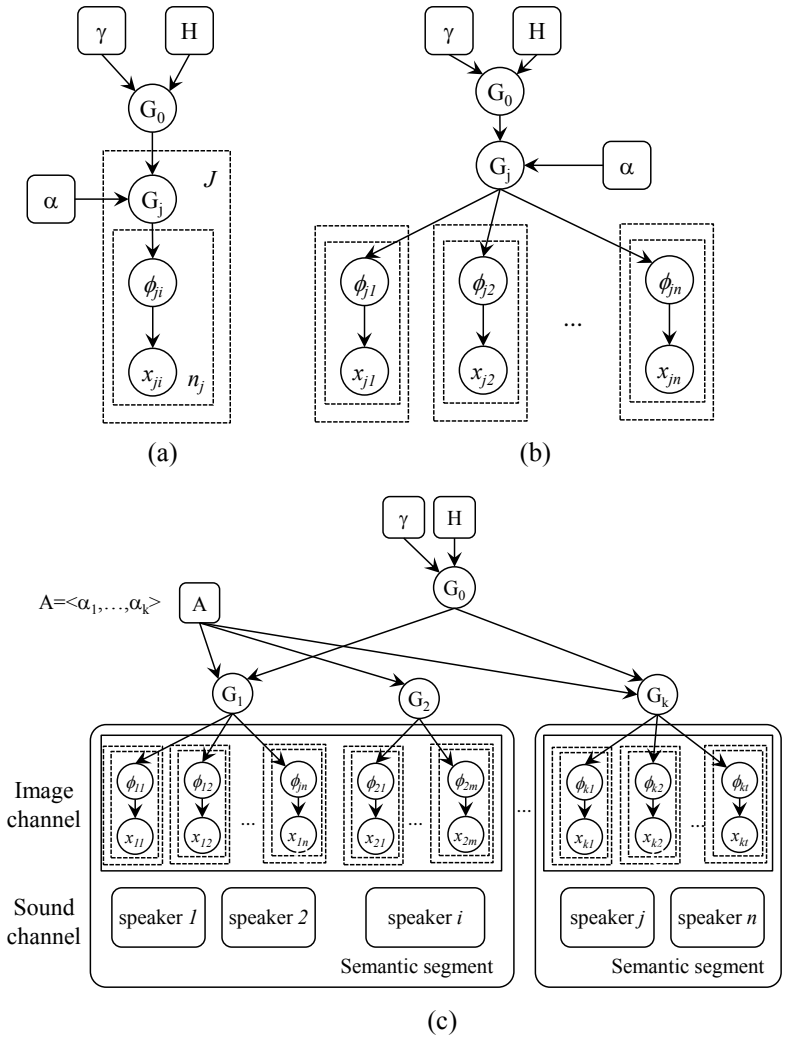


Figure 5.4: Semantic segmentation model: (a) a HDP model. (b) *s*HDP model for a segment (a candidate) (c) Semantic segment model incorporating an image channel and a sound channel.

and backgrounds in an episode. A scene in each video stream corresponds to a table and is divided into groups of conceptual objects. A role of a table (t_{jt}) is to associate a menu (a conceptual object) with customers (visual words) sitting on each table.

Due to its hierarchical structure, HDP is a promising candidate for an image stream analysis. However, HDP cannot cope with abrupt and frequent changes in an image channel due to distortion in light or camera works. Instead of introducing another transition probability like in (Fox et al., 2008), we alleviate abrupt change in a channel by considering changes in other channel. For an image channel, traditional HDP model is constructed. When a new frame is observed, likelihood of the frame by a current HDP model is estimated. If likelihood is lower than a threshold, a new HDP model is estimated (Algorithm 6).

$$G_0|\gamma, H \sim DP(\gamma, H) \tag{5.4}$$

$$G_j|\alpha, G_0 \sim DP(\alpha_0, G_0) \tag{5.5}$$

$$\theta_{ji}|G_i \sim G_i \tag{5.6}$$

Eq. 5.4 $\sim$ Eq. 5.6 explain a typical HDP model. In addition to an image model, changes in a sound channel is considered. A model for a sound channel is explained in section 5.3.3. The combined estimation procedure is explained in section 5.3.4.

5.3.2 Posterior sampling and story change estimation

The sampling method and estimation framework are based on the methods presented in (Ren et al., 2008; Heinrich, 2011). The hyperparameters of this process consist of the baseline probability measure H, and the concentration parameter γ, α. Here, γ and α determine popularity of a table and a story, respectively. We assume gamma

priors for hyperparamters of DPs following an approach presented with HDP (Teh et al., 2005). For posterior inference given $\mathbf{x}$, we need to estimate the change over $(\mathbf{k}, \mathbf{t})$ (where, $k_{t_{db}}$ is an entity (menu) associating a group of visual words and its identifier; $\mathbf{t}$ is a group of visual words).

The intuition behind a menu sampling is that the preferred conceptual objects is likely to retain its popularity. In order to prevent biased preference (only previously served menus are given to a customer), there should be a method to select a new menu. In this work, this intuition follows these observations: (1) a character or background is likely to reappear in a same story; (2) new objects are needed to reconstruct the observed scenes. These observations are represented in the following equations:

$$P(k_{t_{db}} = k|\mathbf{k}^{-tdb}_{\triangle:t-t}) \propto \begin{cases} m_k^{-jt} f_{kt}(x_{jt}) \\ \qquad \text{if } k \text{ is previously used,} \\ \gamma f_{k^{new}}(x_{jt}) \\ \qquad \text{if } k = k^{new}. \end{cases} \tag{5.7}$$

where, m_k^{-jt} denotes the number of parameters associated with $k_{t_{db}}$ and $f_{kt}(x_{ji})$ represents the likelihood of regenerating x_{ji} given k^{new}. $\triangle_{:t-t}$ denotes the duration from the end of previous story to the current scene.

$$P(t_{ji} = t|\mathbf{t}^{-ji}, \mathbf{k}) \propto \begin{cases} n_t f_{kt}(-x_{ji}) \\ \qquad \text{if } t \text{ previously used;} \\ \alpha_0 P(x_{ji}|\mathbf{t}^{-ji}, t_{ji} = t^{new}, \mathbf{k}) \\ \qquad \text{if } t = t^{new}. \end{cases} \tag{5.8}$$

where, n_t represent the popularity of a table t.

5.3.3 Speaker Recognition

The difficulty of a traditional HDP model is that it is only able to explain changes in shots and a semantic segment (scene) could not be described. Contrary to (Ahmed and Xing, 2010) where the prior weight of a component could be defined using a time-decaying kernel with the width and decay factor fixed, it is unrealistic to assume a time-decaying factor in a video stream. Therefore, we incorporate a sound channel using MFCC (Mel Frequency Cepstral Coefficient). With only an image channel or sound channel, there is no safety measure to prevent a premature estimation. However it is possible to ignore an estimated changepoint in a channel if possibility of change in another channel is low. Therefore, it is possible to augment the whole estimation performance by combining independent estimation results in multiple channels.

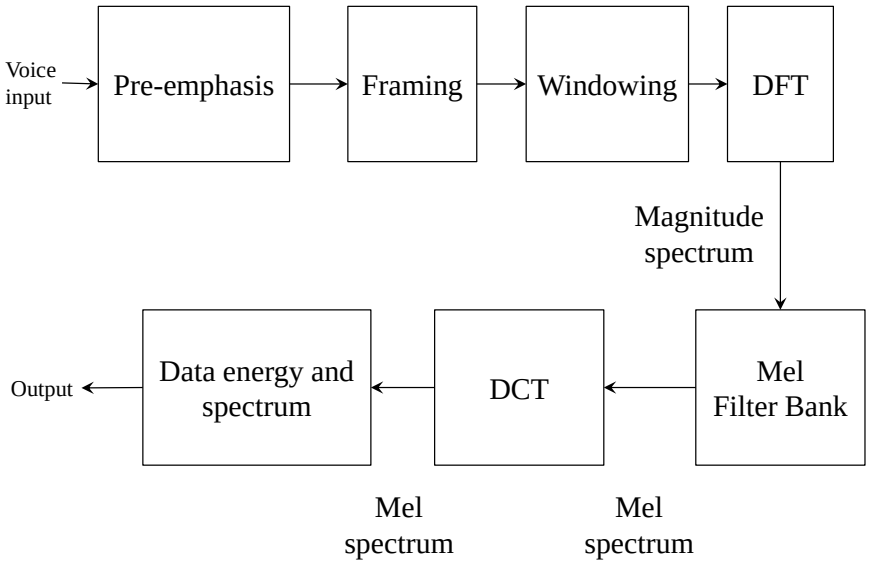


Figure 5.5: MFCC block diagram.

MFCC has been actively used for speaker recognition (Muda et al., 2010; Logan, 2000). Feature extraction process is explained in Fig. 5.5. A sound channel is processed using MFCC to produce voice features. MFCC process is composed of following steps: pre-emphasis, framing, hamming windowing, fast fourier transform, mel filter bank processing, discrete cosine transform, and delta energy and delta spectrum. Each step has its own function and mathematical approaches. We analysis a sound channel in each episode and construct a time interval where a person is speaking.

5.3.4 Dynamic Channel Merging

The proposed method estimates a changepoint by considering a likelihood change measure, $\triangle L_t$ and a dialogue detection measure, $I(S_t)$ (Eq. 5.9). $\triangle L_t$ observes likelihood difference in an image channel and $I(S_t)$ determines whether a dialogue continues.

$$\begin{aligned} F(x_t, s_j|G_L) =& f(\triangle L_t, I(S_j)) \\ =& \omega \cdot \triangle L_t + (1-\omega) \cdot I(S_j) \end{aligned} \tag{5.9}$$

Here,

$$\triangle L_t = \begin{cases} 1 & If\ F(x_t|G_L) - F(x_{t-1}|G_L) < 0 \\ 0 & Else\ F(x_t|G_L) - F(x_{t-1}|G_L) > 0 \end{cases}$$

$$I(S_j) = \begin{cases} 0 & \text{Dialogue} \\ 1 & \text{Silence} \end{cases}$$

$$x_{ji}|\theta_{ji} \sim F_x(\theta_{ji}) \tag{5.10}$$

($F_x(\theta_{ji})$ denotes the likelihood function of x_{ji} given G_j)

$\triangle L_t$ outputs 1 when likelihood of the current frame(x_t) is lower than likelihood of $t-1$th frame. $\triangle L_t$ reflects the insight that a soaring after a plunge in likelihood may be a sign of a new scene. For a sound model, we constructed a model to represent continuity in dialogues. Therefore, Eq. 5.9 describes a criteria that if a likelihood change in an image channel occurs during a dialogue, then dismisses it and if a likelihood change in an image channel co-occurs within a short interval of beginning or end of a dialogue, then considers it as a changepoint candidate.

5.4 Experimental Results

We apply the proposed method on a story changepoints detection problem. In TDT (Topic Detection and Tracking) context, a story is defined as "a topically cohesive segment of news that include two or more declarative independent clauses about a single event." (TDT; Lavrenko et al., 2002). In our approach, a story could be defined as "a topically cohesive segment of episodes that include multiple sentences and events about a single topic." Our task is similar to the segmentation in TDT in terms of segmenting the source stream into its constituent stories (Lavrenko et al., 2002). However, our task is a different challenge because the task is to detect story change in an episode of a TV drama where various characters reappear in limited set of backgrounds. Contrary to our task, the segmentation task in TDT primarily deals with detecting news topic changes. Our approach is similar to the variable-length Markov models (VLMMs) (Liang et al., 2009) because both of our model and VLMMs try to find variable length models capable of interpreting the observed string of symbols. But our model does not assume atomic templates comprising a Markov chain. Our method detects changepoints using only a set of visual words and sound signals. For a story change estimation, a concept of confidence interval could be useful (Brooks et al., 2009). Because the sensitivity and specificity can not

be utilized, we propose a multichannel based model.

5.4.1 Data and Representation

In this section, we illustrate sHDP by measuring its ability to estimate the change of stories in TV series episodes. Human-evaluated ground truth on 3 episodes of a target TV drama is primarily explained in this section.

Data

Table 5.1: Statistical summarization

	Average	Std.	Min	Max
Episode 1	3.87s	2.94s	1.00s	11.00s
Episode 2	3.85s	2.95s	1.00s	11.00s
Episode 3	4.27s	2.86s	1.00s	11.00s
"Std." means "standard deviation"				

19 human participants estimated the change of stories in data manually. Total play time of test material is 125 minutes 30 seconds and we sampled total 7530 scenes to construct test materials. It is very unlikely that several people would estimate a semantic changepoint at the same time. Therefore, an interval of changepoints should be constructed rather than a changepoint. We manually construct an interval by comparing each changepoint and determining a set of changepoints for a similar change. Table 5.1 summarizes statistical characteristics of intervals in each episode. In Table 5.1, minimum length of 1.00s is fine but maximum length of 11.00s seems to be too long. These maximum intervals are corresponding to scenes showing skylines of the background city. In these scenes, it is difficult to select a distinct changepoint.

Representation

Visual words are obtained using SIFT (Scale Invariant Feature Transform) method (Lowe, 1999). A scene of video stream is comprised of image and sound data (in this work, we do not consider text). A sound channel is processed using MFCC. In order to process multi-modal data, other researchers focused on co-occurrence of transient structures in each unimodality (Monach et al., 2009). However, it is difficult to employ similar approach for story transition detection due to the difficulty of devising common structures. Therefore, we combine changes in each channel dynamically rather than utilizing some transient structures.

5.4.2 Story Change Estimation Results

We report the estimated performance of a story change, the characteristics of each story and the trends.

Table 5.2: Proposed method and human evaluations

Human	Episode 1	Episode 2	Episode 3
No. of changes	156 (306)	186(443)	176 (379)
No. of intervals	43	39	39
In this "num1 (num2)" notation, "num1" means # of distinct changes. "num2" is total # of the evaluated changes.			

Experimental results

We report performance of the proposed method in terms of Precision, recall, and F_1 measures in Table 5.2, Table 5.3, and Table 5.4.

Table 5.3: Best precision

Computer	Episode 1	Episode 2	Episode 3
No. of changs	8	6	9
Precision	0.62	0.38	0.82
Recall	0.15	0.13	0.20

Table 5.4: Best recall

Computer	Episode 1	Episode 2	Episode 3
No. of changs	52	45	41
Precision	0.059	0.060	0.057
Recall	0.96	0.96	0.93

$$\text{Precision} = \frac{\#(\text{Correctly estimated changepoints})}{\#(\text{Total estiamted changepoints})}$$

$$\text{Recall} = \frac{\#(\text{Correctly estimated changepoints})}{\#(\text{Total ground truth intervals})}$$

$$F_1 = \frac{2 \times Recall \times Precision}{Recall + Precision}$$

5.4.3 Comparison with Other Method

In (Parshyn and Chen, 2006), the authors proposed a method based on video coherence and audio dissimilarity. Among the methods introduced in (Parshyn and Chen, 2006), we compare the HMM-based segmentation method as this method does not utilizes heuristic rules. Among the experimental materials, we select "A

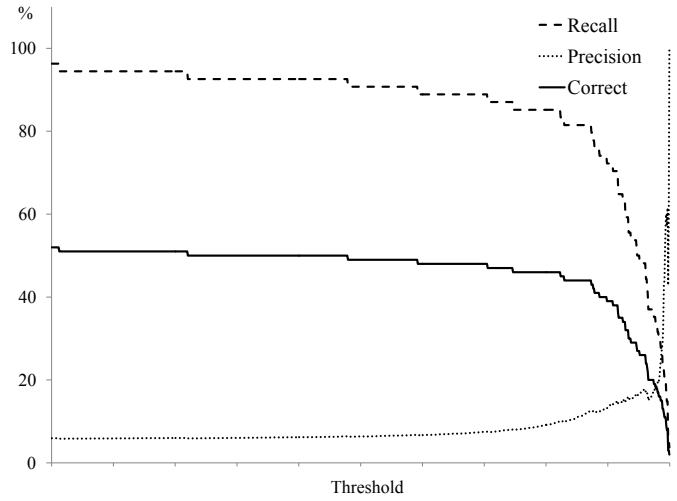


Figure 5.6: Recall, precision, and the number of correctly estimated changepoints ("correct") of episode 1.

Table 5.5: Performance Comparison

	The proposed method	HMM-based method
Precision	0.41	0.55
Recall	0.73	0.72
F_1	0.53	0.62

beautiful mind" and prepare ground-truth data based on evaluation results by 10 human participants.

Although the proposed method reports inferior results, these results are promising. Our method employs very basic framework. An image channel is processed using HDP method without utilizing object recognition techniques. A sound chan-

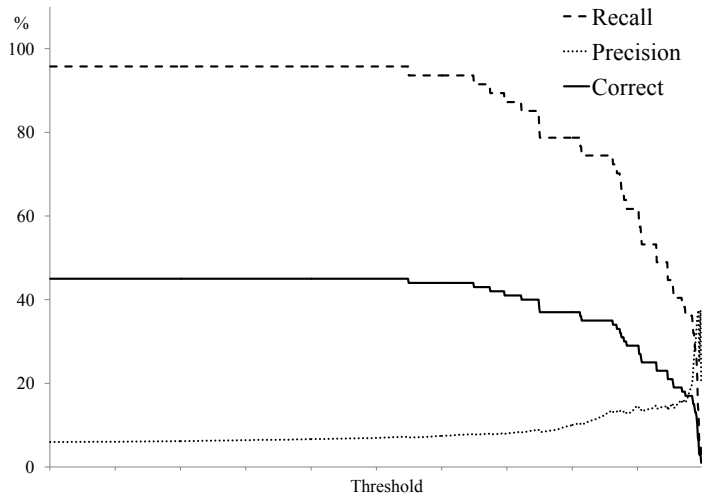


Figure 5.7: Recall, precision, and the number of correctly estimated changepoints ("correct") of episode 2.

nel has much more room for improvement. After MFCC processing, only clustering is applied to build groups of similar vectors. With appropriate similarity measure, the segmentation performance could be improved.

5.5 Discussion and Summarization

In this paper, we have introduced a semantic segmentation scheme with two central components: multichannel analysis and dynamic channel merging. We proposed a sequential HDP model to utilize inherent hierarchies in an image channel and speaker recognition based interval estimation model for a sound channel. Although abrupt changes are observed in each channel, we have demonstrated high-performance scene change detection by combining estimation in multiple channels.

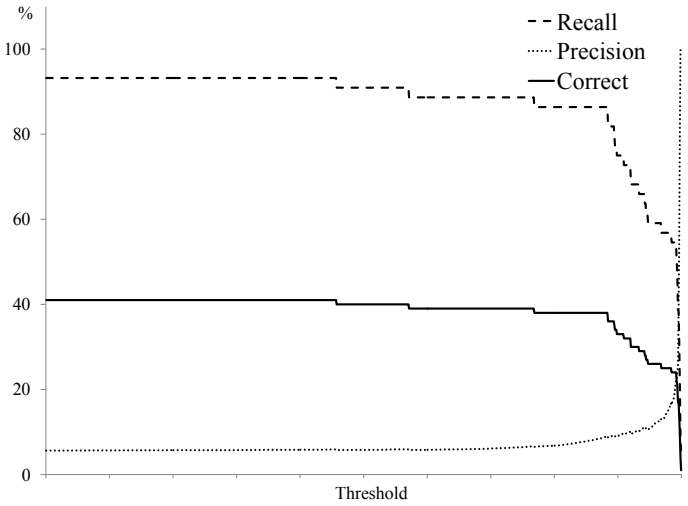


Figure 5.8: Recall, precision, and the number of correctly estimated changepoints ("correct") of episode 3.

There are several possible directions for future works. Algorithmic efficiency could be significantly improved by adopting an object recognition engine and constructing a object dictionary for an episode. More complete feature selection can also be performed to realize more succinct representation. Finally, we believe this method can be applied to more ambitious goal of contents recommendation by developing a segment descriptor based on estimated distributions in each modality.

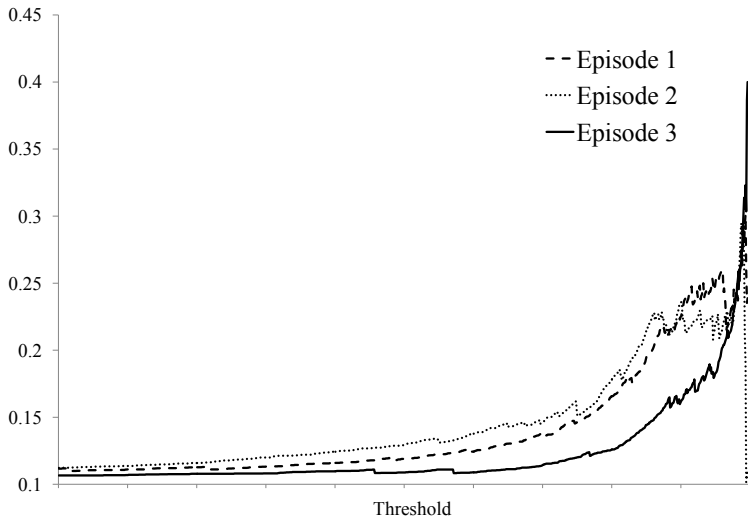


Figure 5.9: F1 measure of episode 1, 2, 3.

Chapter 6

Concluding Remarks

We begin with a summary of the discussed themes and principal contributions in the preceding chapters. Throughout these chapters, we developed a feature relevance network for indoor location estimation, an evolutionary particle filtering based method for learning inherent dependencies in a temporal stream, and a nonparametric Bayesian framework to describe semantic changes in a multi-modal stream.

6.1 Summary of Methods and Contributions

As we have demonstrated in this thesis, there exist useful cues for analyzing dynamical complex phenomena despite of uncertainty in the underlying dynamics of these phenomena.

When underlying distributions are changing, invariant properties provide a key for analyzing datasets. For efficient processing of changing distributions, we proposed a feature relevance network framework. The proposed method is composed of mapping, structure expansion, and estimation steps. Through these steps, a feature relevance network $\mathcal{Q}$ is obtained and given data points are expanded until being

converged to the same structure. By building a feature relevance network based on invariant properties, we were able to map given instances onto a new problem space and make a cluster of adjacent data instances. For comparing adjacent data points, a novel concept of expansion cost is introduced to represent distance between data points indirectly through difficulties in expansion. The usefulness of the proposed method was verified on the indoor location estimation problem. This is the first contribution of this thesis. When it is nearly impossible to assume any prior knowledge due to unpredictable changes, typical machine learning methods fail to provide appropriate schemes for this kind of problems. The proposed method provided an insight for learning in a uncertain world by transferring problem space based on invariant properties.

Typical approaches for temporal streams have employed inflexible assumptions such as repeating frames or fixing underlying distributions. However, these assumptions on the underlying distributions and associated parameters are unrealistic to deal with temporal streams such as TV dramas. In order to analyze temporal streams without adopting rigid assumptions, we proposed an evolutionary particle filtering based sequence learning scheme. The proposed learning scheme is composed of a segmentation step and a dependency learning step. In the segmentation step, a population of particles represent an image segment in a collaborative manner and this population is used to estimate changes in a given temporal stream. After segmenting, multiple populations are generated. Through dependency learning, inherent relations between segment are extracted and the transitional probability between segments is computed. Because the proposed scheme retains dominant features of a segment, it is possible to utilize the proposed method as a compression method for a temporal stream. We applied the proposed method on the problem of detecting dominant image changes in episodes of a TV drama. This is the second

contribution of this thesis. We showed that it is possible to analyze a temporal stream effectively by employing a collaborative representation of particles. Through collaborative representation, the proposed method was able to focus on dominant features while overcoming minor distortions due to lightings or camera-works. By employing an evolutionary approach, the proposed method overcame the difficulty of premature convergence and introduced essential diversity. Experimental results showed that an EM (Expectation-Minimization) approach with a population based representation could provide a clue for temporal stream processing.

The final contribution of this thesis is a use of hierarchical structures inherent in temporal streams. In a TV drama, we usually encounter various hierarchies of visual words, objects, and a higher set of objects or hierarchies of words, phrases, and sentences. In order to exploiting these inherent hierarchies, we took a Bayesian nonparametric approach of hierarchical Dirichlet process (HDP). We constructed a latent model for each segment in an image channel using HDP. A sound channel is processed by building dialogue intervals. By merging estimations in an image channel and a sound channel, we were able to approximate semantic changes in a TV drama without employing a semantic dictionary that would be costly to build. Experimental results showed that a nonparametric approach focusing on basic segments can process a temporal stream.

These three contributions of this thesis are not distinct ones. Each individual research was an attempt to answer the question of "how to cope with real-world phenomena without useful but restricting assumptions on the target domains". From Chapter 3 and Chapter 4, we showed that invariant feature relations in a domain and a population based representation provide clues for processing target domains without assuming prior knowledge. In Chapter 5, we showed that a Bayesian nonparametric approach could yield reasonable inference based on flexible models.

6.2 Suggestions for Future Research

There are several possible directions for future works by our approaches. Each of the preceding chapter has presented a promising application domain. We additionally present a few new future research directions.

Analysis of Contents

we suggested a learning method focusing on invariant relations in Chapter 3. Although at a conceptual level, there exist invariant relations between characters or between characters and backgrounds of video streams. If various conceptual objects are recognized, the proposed method in Chapter 3 can be used to analyze interactions between conceptual objects.

Contents Recommendations

In Chapter 4 and Chapter 5, we introduced methods for temporal streams analysis. These methods could be basic researches for stream recommendation. With a latent model for a semantic segment, it is possible to assign a semantic descriptor for each segment based on the characteristics of its image channel and sound channel. If we have a database of semantic descriptors, it is possible to compare semantic descriptors of a user interested stream with descriptor for other materials and recommend materials based on the comparison results.

Bibliography

A. Ahmed and E. P. Xing. Dynamic non-parametric mixture models ans the recurrent chinese restaurant process: with applications to evolutionary clustering. In *The 8th SIAM International Conference on Data Mining*, pages 219–230, Atlanta, Georgia, April 2008.

A. Ahmed and E. P. Xing. Timeline: A dynamic hierarchical dirichlet process model for recovering birth/death and evolution of topics in text stream. In *The 26th Conference on Uncertainty in Artificial Intelligence*, pages 20–29, Catalina Island, California, July 2010.

F. J. Ambrosio and F. M. Kinniry Jr. Stock market volatility measures in perspective. Technical report, Vanguard Investment Counseling & Researchg, 2008.

R. K. Ando and T. Zhang. A framework for learnig predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, 6: 1817–1853, 2005.

M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2):174–188, 2002.

ATR. URL `http://en.wikipedia.org/widi/Average_True_Range`.

G. A. Barreto and A. F. R. Araújo. Unsupervised learning and recall of temporal sequences: an application to robotics. *Internaional Journal of Neural Systems*, 9 (3):235 – 242, 1999.

M. Belkin and P. Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *Advances in Neural Information Processing Systems*, pages 585–591, Vancouver, B.C, December 2001. MIT Press.

S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira. Analysis of representation for domain adaption. In *Advances in Neural Information Systems*, pages 137–144, Vancouver, B.C, December 2007. MIT Press.

D. Blackwell and J. MacQueen. Ferguson distribution via pólya urn schemes. *Annals of Statistics*, 1:353–355, 1973.

D. Borsboom, G. J. Mellenbergh, and J. Heerden. The theoretical status of latent variables. *Psychological Review*, 110(2):203–219, 2003.

R. R. Brooks, J. M. Schwier, and C. Griffin. Behavior detection using confidence intervals of hidden markov models. *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics*, 39(6):1484 – 1492, 2009.

D. Cai, X. Wang, and X. He. Probabilistic dyadic data analysis with local and global consistency. In *The 26th International Conference on Machine Learning*, pages 105–112, Montreal, Quebec, June 2009.

K. R. Canini and T. L. Griffiths. A nonparametric bayesian model of multi-level category learning. In *Proceedings of the 25th AAAI Conference on Artificial Intelligence (AAAI)*, San Francisco, Calif., August 2011. AAAI Press.

F. Caron, M. Davy, and A. Doucet. Generalized polya urn for time-varying dirichlet

process mixtures. In *The 23rd Conference on Uncertainty in Artificial Intelligence*, pages 33–40, Vancouver, B. C., July 2007.

R. Caruana. Multitask learning. *Machine Learning*, 28:41–75, 1997.

L. Chaisorn, T.-S. Chua, and C.-H. Lee. A multi-modal approach to story segmentation for new video. *World Wide Web: Internet and Web Information Systems*, 6(2):187–208, 2003.

V. Chasanis, A. Kalogeratos, and A. Likas. Movie segmentation into scenes and chapters using locally weighted bag of visual words. In *Proceedings of the ACM International Conference on Image and Video Retrieval*, Santorini, Greece, July 2009.

C. Chen, J. Zhang, and R. Fleischer. Distance approximating dimension reduction of riemannian manifolds. *IEEE Transactions on Systems, Man, and Cybernetics, - Part B: Cybernetics*, 40(1):208 – 217, 2010.

T. W. S. Chow, W. Piyang, and E. W. M. Ma. A new feature selection scheme using a data distribution factor for unsupervised nominal data. *IEEE Transactions on Systems, Man, and Cybernetics, - Part B: Cybernetics*, 38(2):499 – 509, 2008.

P.-A. Coquelin, R. Deguest, and R. Munos. Particle filter-based policy gradient in pomdps. In *Advances in Neural Information Processing Systems 21*, Vancouver, B.C, December 2009. MIT Press.

T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*. MIT Press, Cambridge, MA, 2001.

W. Dai, G.-R. Xue, Q. Yang, and Y. Yu. Transferring naive bayes classifiers for text classification. In *The 22rd AAAI Conference on Artificial Intelligence*, pages 540–545, Vancouver, B.C, July 2007. MIT Press.

K. Deb, S. Gupta, D. Daum, J. Branke, A. K. Mall, and D. Padmanabhan. Reliability-based optimization using evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, 13(5):1054 – 1074, 2009.

C. B. Do and A. Y. Ng. Transfer learning for text classification. In *Advances in Neural Information Processing Systems*, pages 299–306, Vancouver, B.C, December 2006. MIT Press.

R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. John Wiley and Sons, Inc., New York, NY, 2001.

D. B. Dunson. Bayesian dynamic modeling of latent trait distributions. *Biostatistics*, 7:551–568, 2006.

T. Evgeniou and M. Pontil. Regularized multi-task learning. In *The 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 109–117, Seattle, W. A., August 2004.

P. Fearnhead. Exact and efficient bayesian inference for multiple changepoint problems. *Statistics and Computing*, 16:203 – 213, 2006.

T. Ferguson. A bayesian analysis of some nonparametric problems. *Annals of Statistics*, 1(2):209–230, 1973.

E. B. Fox. *Bayesian Nonparametric Learning of Complex Dynamical Phenomena*. PhD thesis, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, 2008.

E. B. Fox, E. B. Sudderth, M. I. Jordan, and A. S. Willsky. An hdp-hmm for systems with state persistence. In *The 25th International Conference on Machine Learning*, pages 312–319, Helsinki, Finland, July 2008.

S. J. Gershman, D. M. Blei, and Y. Niv. Context, learning, and extinction. *Psychological Review*, 117(1):197–209, 2010.

Z. Ghahramani. Non-parametric bayesian methods. *Uncertainty in Artificial Intelligence Tutorial*, 2005.

Y. Gwon, R. Jain, and T. Kawahara. Robust indoor location estimation of stationary and mobile users. In *The 24th IEEE INFOCOM*, pages 1734–1743, Miami, Florida, March 2004.

G. Heinrich. Infinite lda - implementing the hdp with minimum code complexity. Technical report, arbylon.net, 2011. Technical Note TN2011/1.

G. Heitz, G. Elidan, and D. Koller. Transfer learning of object classes: from cartoons to photographs. In *NIPS 2005 Workshop - Inductive Transfer: 10 Years Later*, Vancouver, B. C., December 2005.

T. Higuchi. Monte carlo filter using the genetic algorithm operators. *Journal of Statistical Computation and Simulation*, 59(1):1–23, 1997.

J. Huang, A. Smola, A. Gretton, K. M. Borgwardt, and B. Schölkopf. Correctign sample selection bias by unlabeled data. In *Advances in Neural Information Systems*, pages 601–608, Vancouver, B.C, December 2007. MIT Press.

Z. Ibrahim, P. Gros, and S. Campion. Avsst: an automatic video stream structuring tool. In *The 3rd Networked and Electronic Media Summit*, Barcelona, Spain, October 2010.

ICDM-web. URL `http://www.cse.ust.hk/~qyang/ICDMDMC07/`.

H. Daumé III and D. Marcu. Domain adaptation for statistical classifiers. *Journal of Artificial Intelligence Research*, 26:101–126, 2006.

T. Jebara. Multi-task feature and kernel selection for svms. In *The 21st International Conference on Machine Learning*, Banff, Alberta, July 2004.

K. Kaemarungsi and P. Krishnamurthy. Modeling of indoor positioning systems based on location fingerprinting. In *INFOCOM 2004*, pages 1012–1022 vol. 2, March 2004.

J. Kennedy and R. Eberhart. Particle swarm optimization. In *IEEE International Conference on Neural Networks*, pages 1942 – 1948, Perth, Western Australia, November-December 1995. IEEE Neural Networks Council.

I. Koprinska and S. Carrato. Temporal video segmentation: a survey. *Signal Processing: Image Communication*, 16:477 – 500, 2001.

K. Krawiec and Bir Bhanu. Visual learning by evolutionary and coevolutionary feature synthesis. *IEEE Transactions on Evolutionary Computation*, 11(5):635 – 650, 2007.

J. B. Kruskal and M. Liberman. The symmetric time-warping problem: from continuous to discrete. In D. Sankoff and J. B. Kruskal, editors, *Time Warps, String Edits, and Macromolecules - The Theory and Practice of Sequence Comparison.* CSLI Publications, Stanford, CA, 1999.

N. M. Kwok, G. Fang, and W. Zhou. Eolutionary particle filter: re-sampling from the genetic algorithm perspective. In *IEEE/RSJ International Conference on Intelligent Robots and Systems 2005*, pages 2935 – 2940, Alberta, August 2005.

T. Lane and C. E. Brodley. Temporal sequence learning and data reduction for anomaly detection. *ACM Transactions on Information and System Security*, 2 (3):295 – 331, 1999.

V. Lavrenko, J. Allan, E. DeGuzman, D. LaFlamme, V. Pollard, and S. Thomas. Relevance models for topic detection and tracking. In *The 2nd International Conference on Human Language Technology Research*, pages 115 – 121, San Diego, California, March 2002. Morgan Kaufmann.

N. D. Lawrence and J. C. Platt. Learning to learn with the informative vector machine. In *The 21st International Conference on Machine Learning*, pages 512–519, Banff, Alberta, July 2004.

S.-I. Lee, V. Chatalbashev, D. Vickrey, and D. Koller. Learning a meta-level prior for feature relevance from multiple related tasks. In *The 24th International Conference on Machine Learning*, pages 489–496, Corvallis, Oregon, June 2007.

R. Levy, F. Reali, and T. L. Griffiths. Modeling the effects of memory on human online sentence processing with particle filters. In *Advances in Neural Information Processing Systems 21*, pages 937–944, Vancouver, B.C, December 2009. MIT Press.

Y.-M. Liang, S.-W. Shih, A. C.-C. Shih, H.-Y. M. Liao, and C.-C. Lin. Learning atomic human actions using variable-length markov models. *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics*, 39(1):268 – 280, 2009.

H. Liu, H. Darabi, P. Banergee, and J. Liu. Survey of wireless indoor positioning techiques and systems. *IEEE Transactions on System, Man, and Cybernetics-Part C: Applications and Reviews*, 37(6):1067–1080, 2007.

B. Logan. Mel frequency cepstral coefficients for music modeling. In *International Symposium on Music Information Retrieval*, Plymouth, Massachusetts, October 2000.

D. G. Lowe. Object recognition from local scale-invariant features. In *The Proceedings of the Seventh IEEE International Conference on Computer Vision*, pages 1150 – 1157, Kerkyra, Greece, September 1999. IEEE Computer Society.

G. Manson and S.-A. Berrani. Automatic tv broadcast structuring. *International Journal of Digital Multimedia Broadcasting*, 2010:Article ID 153160, 2010.

A. Mittal. An overview of multimedia content-based retrieval strategies. *Informatica*, 30:347–356, 2006.

G. Monach, P. Vandergheynst, and F. T. Sommer. Learning bimodal structure in audio-visual data. *IEEE Transactions on Neural Networks*, 20(12):1898–1910, 2009.

L. Muda, M. Begam, and I. Elamvazuthi. Voice recognition algorithms using mel frequency cepstral coefficient (mfcc) and dynamic time warping (dtw) techniques. *Journal of Copmuting*, 2(3):138 – 143, 2010.

M. Mühlich. Particle filters an overview. Technical report, Internaional Summer School in Brasov/Romania, 2003.

L. Murray and A. Storkey. Continuous time particle filtering for fmri. In *Advances in Neural Information Processing Systems 20*, Vancouver, B.C, December 2008. MIT Press.

J. Namikawa and J. Tani. A model for learning to segment temporal sequences, utilizing a mixture of rnn experts together with adaptive variance. *Neural Networks*, 21:1466 – 1475, 2008.

A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42(3):145 – 175, 2001.

P. Orbanz and Y.-W. Teh. Bayesian nonparametric models. In S. Claude and W. Geoffrey, editors, *Encyclopedia of Machine Learning*. Springer, New York, USA, 2010.

P. Orbanz, S. Braendle, and J. M. Buhmann. Bayesian order-adaptive clustering for video segmentation. In *Energy Minimization Methods in Computer Vision and Pattern Recognition*, pages 334–349, Ezhou, China, August 2007.

J. J. Pan, J. T. Kwok, Q. Yang, and Y. Chen. Accurate and low-cost location estimation using kernels. In *The 19th International Joint Conference on Artificial Intelligence*, pages 1366–1371, Edinburgh, Scotland, August 2005.

S. J. Pan and Q. Yang. A survey on transfer leaning, 2008. URL `http://www.cse.ust.hk/~sinnopan/SurveyTL.htm`.

Y. Pan and S. A. Billings. Neighborhood detection for the identification of spatiotemporal systems. *IEEE Transactions on Systems, Man, and Cybernetics, - Part B: Cybernetics*, 38(3):846 – 854, 2008.

S. Park, J. P. Hwang, E. Kim, and H.-J. Kang. A new evolutionary particle filter for the prevention of sample impoverishment. *IEEE Transactions on Evolutionary Computation*, 13(4):801 – 809, 2009.

V. Parshyn and L. Chen. Video segmentation into scenes using stochastic modeling. Technical report, Ecole Centrale de Lyon, 2006. Research Report.

E. Pekalska and R.P.W. Duin. Beyond traditional kernels: classification in two dissimilarity-based representation spaces. *IEEE Transactions on Systems, Man, and Cybernetics, - Part C: Applications and Reviews*, 38(6):729–744, 2008.

M. K. Pitt. Smooth particle filters for likelihood evaluation and maximisation.

Technical report, The University of Warwick, 2002. Technical Note, Department of Economics.

J.-P. Poli. An automatic television stream structuring system for television archives holders. *Multimedia Systems*, 14:255–275, 2008.

R. J. Radke, S. Andra, O. Al-Kofahi, and B. Roysam. Image change detection algorithms: a systematic survey. *IEEE Transactions on Image Processing*, 14(3): 294 – 307, 2005.

R. Raina, A. Battle, H. Lee, B. Packer, and A. Y. Ng. Self-taught learning: transfer learning from unlabeled data. In *The 24th International Conference on Machine Learning*, pages 759–766, Corvallis, Oregon, June 2007.

I. M. Rekleitis. A particle filter tutorial for mobile robot localization. Technical report, Centre for Intelligent Machines, McGill University, 2004. Technical Report.

L. Ren, L. Carin, and D. B. Dunson. The dynamic hierarchical dirichlet process. In *The 25th International Conference on Machine Learning*, pages 824–831, Helsinki, Finland, July 2008.

J. Sethuraman. A constructive definition of dirichlet priors. *Statistica Sinica*, 4: 639–650, 1994.

P. Sidiropoulos, V. Mezaris, I. Kompatsiaris, H. Meinedo, M. Bugalho, and I. Trancoso. Temporal video segmentation to scenes using high-level audiovisual features. *IEEE Transactions on Circuits and Systems for Video Technology*, 21(8):1163 – 1177, 2011.

TDT. URL `http://projects.ldc.upenn.edu/TDT/`.

Y. W. Teh and M. I. Jordan. Hierarchical bayesian nonparametric models with applications. In N. Hjort, C. Holmes, P. Mueller, and S. Walker, editors, *Bayesian Nonparametrics: Principles and Practice*. Cambridge University Press, Cambridge, UK, 2010.

Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei. Hierarchical dirichlet processes. Technical report, National University of Singapore, 2005. Technical Report, Deptment of Computer Science.

R. van der Merwe, A. Doucet, and E. Wan N. de Freitas. The unscented particle filter. In *Advances in Neural Information Processing Systems*, Denver, Colorado, December 2000. MIT Press.

N. Wagner, Z. Michalewicz, and R. R. McGregor. Time series forecasting for dynamic environments: The dyfor genetic program model. *IEEE Transactions on Evolutionary Computation*, 11(4):433 – 452, 2007.

P. Wang, C. Domeniconi, and K. B. Laskey. Nonparametric bayesian clustering ensembles. In *European Conference on Machine Learning and Priniples and Practice fo Knowledge Discovery*, pages 435–450, Barcelona, Spain, September 2010.

X. Wang and A. McCallum. Topics over time: A non-markov continuous-time model of topical trends. In *The 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 424–433, Pholadelphia, USA, August 2006.

Y. Wang, L. Zhou, J. Feng, and J. Wang. Mining complex time-series data by learning markovian models. In *The 2006 International Conference on Data Mining*, pages 1136–1140, Las Vegas, Nevada, June 2006.

Z. Wang, Y. Song, and C. Zhang. Transferred dimensionality reduction. In *The European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 550–565, Antwerp, Belgium, September 2008.

X. Wei, J. Sun, and X. Wang. Dynamic mixutre models for multiple time series. In *The 20th International Joint Conference on Artififical Intelligence (IJCAI-07)*, pages 2909–2914, Hyderabd, India, January 2007.

L. Xie, L. Kennedy, S.-F. Chang, A. Divakaran, H. Sun, and C.-Y. Lin. Layered dynamic mixture model for pattern discovery in asynchronous multi-modal streams. In *IEEE International Conference on Acoustics, Speech, and Signal Processing 2005 (ICASSP '05)*, pages ii/1053–ii/1056, Philadelphia, Pennsylvania, March 2005.

X. Xuan and K. Murphy. Modeling changing dependency structure in multivariate time series. In *International Conference on Machine Learning 2007*, pages 1055 – 1062, Corvalis, Oregon, June 2007. Omni Press.

Q. Yang, S. J. Pan, and V. W. Zheng. Estimating location using wi-fi. *IEEE Intelligent Systems*, 23(1):8–13, 2008.

J. Yin and Q. Yang. Learning adaptive temporal radio maps for signal-strength-based location estimation. *IEEE Transactions on Mobile Computing*, 7(7):869–883, 2008.

B.-T. Zhang. Hypernetwork: a molecular evolutionary architecture for cognitive learning and memory. *IEEE Computational Intelligence Magazine*, 3(3):49 – 63, 2008.

D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf. Learning with

local and global consistency. In *Advances in Neural Information Systems*, pages 321–328, Vancouver, B.C, December 2004. MIT Press.

Q. Zhu and Z. Song. Dynamic video segmentation via a novel recursive bayesian learning method. In *17th IEEE International Conference on Image Processing*, pages 2997 – 3000, Hong Kong, September 2010.

초 록

시간성을 띄고 있는 복잡한 스트림 데이터를 효율적으로 처리하려면 스트림 데이터를 구성하는 요소간의 은닉 의존성을 고려할 필요가 있다. 그러나 스트림을 구성하는 각 모달리티(modality)에서의 급격한 변화 및 스트림 데이터에 내제된 계층적 특성과 같은 어려움 때문에 은닉 의존성 분석은 매우 어려운 작업이다. 본 논문에서는 시간성을 띄고 있는 스트림 데이터에 내재된 의존성을 추출하고 해당 환경에서의 모델링과 추론을 가능하게 하기 위해 다음과 같이 세 가지 알고리즘을 개발하였다. 첫째, 기저 분포가 변하는 환경에서 대응할 수 있는 추론 알고리즘을 제안하였다. 둘째, 문제 공간의 요약 자료 구조를 생성하면서 시간성을 띄고 있는 스트림을 학습하는 알고리즘을 개발하였다. 마지막으로 복수 개의 양상으로 구성된 스트림 처리를 위한 알고리즘을 개발하였다.

시간성을 고려할 경우 환경 원인 때문에 안정된 기저 분포를 가정할 수 없는 문제 도메인이 존재한다. 이와 같은 도메인에서 분포 변화를 추정할 수 있도록 본 논문에서는 특성 관계 네트워크에 기반한 추론 알고리즘을 개발하였다. 데이터의 특성에 따라 환경 변화가 발생해도 변하지 않는 관계가 존재하는데, 우리는 이 관계성에 주목하여 특성 관계 네트워크라는 개념을 제안하였다. 제안 알고리즘은 무선 신호에 기반한 실내 위치 추정 문제에 적용되었으며, 특성 분포를 사전에 가정하지 않는 비모수적 접근을 이용하여 특성 간에 숨겨진 관계를 찾을 수 있음을 확인하였다.

시간성을 띄고 있는 스트림 데이터의 가장 대표적인 예는 TV 드라마와 같은 멀티미디어 동영상이다. TV 드라마의 에피소드에 숨겨진 의존성을 추정하고 새롭고 간결한 형태로 전체 에피소드를 나타낼 수 있도록 본 논문에서는 파티클 집합에 기반한 의존관계 학습 방법을 개발하였다. 제안 방법은 사전 분포를 가정하는 대신 파티클이 협동적인 방식으로 주어진 스트림의 우세 특성을 포착한다는 점을 특징으로 한다. 제안 방법에서는 2단계 학습을 통해 파티클을 진화시키는데 첫 번째 단

계에서는 진화 파티클 필터링(Evolutionary particle filtering)을 이용하여 우점 영상 세그먼트를 추정하며, 두 번째 단계에서는 추정된 세그먼트 사이의 의존성을 나타내는 전이확률매트릭스를 구축한다. 우리는 인간 평가자에 의한 평가 결과와 제안 방법의 세그먼트 구분 관계를 비교하고 시드 이미지(Seed image)가 주어졌을 때 예상되는 세그먼트 스트림을 생성하여 제안 방법의 성능을 보였다.

마지막으로 영상과 소리가 복합된 멀티채널 환경에서의 추론을 위하여 순차적 계층 Dirichlet 프로세스(Sequential hierarchical Dirichlet process, sHDP) 모델을 개발하였다. 멀티채널 데이터의 원활한 처리를 위하여 sHDP는 멀티채널 스트림을 단일 양상으로 구성된 하위 채널로 분할한 후 각 하위 채널에 대해 은닉 변수 모델을 추정한다. 전체 스트림에서의 변화는 각 하위 채널에서의 변화를 동적으로 병합하여 추정하였다. 제안 방법을 실제 TV 드라마 에피소드의 스토리 변화 추정 문제에 적용하였으며, 인간 평가자의 평가 결과와 비교하여 성능을 확인하였다.

데이터에 내재된 의존성을 통합함으로써 우리는 시간성을 띄고 있는 스트림 데이터에 적용할 수 있는 알고리즘을 개발하였다. 제안된 방법론은 다양한 응용 분야에서 뛰어난 추정 능력을 보임으로써, 시간 측도에서 지속적으로 변화하는 데이터 분석에 효과적임을 입증하였다.

주요어: 확률모델, 비모수적 방법론, 시공간적 패턴,
진화 파티클 필터링, 스트림 분석

학번: 2004-31034