Service Pattern Evaluation: Studying Profitability from Perspective of Resource

Zhiling Luo
Computer Science
Zhejiang University
Hangzhou, China
Email: luozhiling@zju.edu.cn

Ying Li
Computer Science
Zhejiang University
Hangzhou, China
Email: cnliying@zju.edu.cn

Jianwei Yin
Computer Science
Zhejiang University
Hangzhou, China
Email: zjuyjw@cs.zju.edu.cn

Honghao Gao
Computer Science
Zhejiang University
Hangzhou, China
Email: gaohonghao@zju.edu.cn

Yuyu Yin
Computer Science
Hangzhou Dianzi University
Hangzhou, China
Email: yinyuyu@hdu.edu.cn

***Abstract*—The profitability, instead of technical satisfaction, becomes the core factor in service designing for Modern Service Companies. In this paper, a service pattern evaluation framework is introduced for evaluating the service's profitability service designing. Our framework takes two phases: the service pattern with resource usage situation is extracted by the service description and system log. For each resource, we study the circulation of resource by estimating the probability that the creating amount can satisfy the requirement. Besides, the funds and money, as a special resource, is studied to estimate the profitability of service. The experiment results on the real and synthetic datasets, report the efficiency and effectiveness of our approach.**

***Keywords*-Service Pattern; Service Pattern Evaluation; Resource Recycling; Probability Model; Profitability Evaluation**

I. INTRODUCTION

The increase value of global modern service industry occupies over 60% of total yield value, major developed countries share over 70%, even if the moderate and low-income countries reach the average level of 45% [1]. Industrial structure has been transformed from industrial-based economy" to "service-based economy,". Mobile Application, E-Business and online services are typical modern services [2][3]. However, there are too many modern service companies providing good services in technical while failed to change them into profit. For example, in 2013, there are over 1 million Mobile Applications on Apple APP Store, while only 5% developers earned money. A few hot APPs attracted a huge number of customers and made mass bulks. For example, in 2014, CLASH of CLANS, an online multi-player mobile game, supporting both Android and IOS platform, welcomed 39 thousand new players and earned 1.11 Million dollar per day. It comes a problem: **what factors do affect the services' profitability** (Q1). The further problems include **how to quantitatively analyze these factors** (Q2) and **how to re-organize or re-create service for better profitability** (Q3).

Among these three problems, the first one, Q1 is the most basic and essential for both service community researchers and service managers. Existing works on evaluating service most concentrate on the technical parts, for example, reliability and response time. These criteria are called Quality of Service, QoS for short [4]. Given the QoS of atomic services, the estimation of QoS on whole service process is called QoS prediction [5] which mainly formulated by multiple attribute decision analysis (MADA) [6]. This framework is reasonable while not enough for profitability analyzing problem, namely Q1, because the QoS satisfaction ensures the service works well in function instead of making good profit.

Example I.1. *Youtube, the famous and widely used online video playing website, earns millions of dollars by selling its advertisement position. There are two essential resources contributing to its business pattern: the video and customer traffic. The digital video, uploaded by the customers is the production which attracts more and more customers to visit Youtube. With so many people visiting, the customer traffic, is also become valuable for advertisers, which put Ad. on Youtube and pay for each watching.*

From this example, we can see that it is the resource, namely video and customer traffic, rather than technical function that mainly contribute to Youtube's profitability. Therefore, for better studying Q1, we introduce the **resource circulation** as a key factor in analyzing the service profitability. When we say circulation, we mean the procedure with resource **producing** and **consuming**. Intuitively speaking, the resource should be created in some activities and consumed in other activities. The basic principle is that the well-designed service should ensure that the producing rate is larger than the consuming rate. However, in practice, the producing and consuming amount is variant in different situation. For example, an advertiser may pay 1 dollar for each Ad. watching in 2015 while 0.8 dollar in 2016.

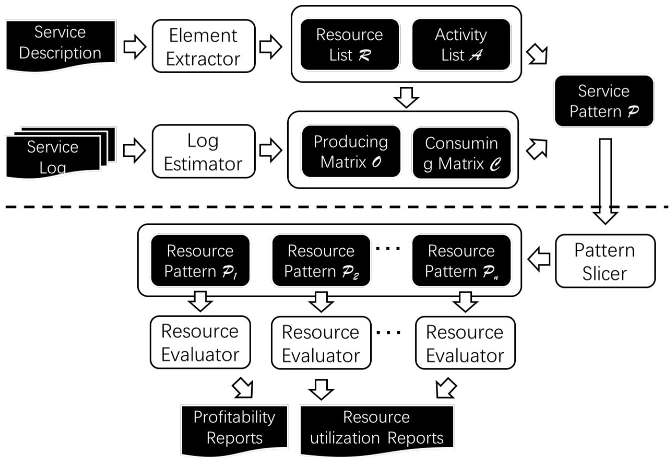


Figure 1: Service Pattern Evaluation framework.

Therefore, in this paper, we introduce a probability model, in which the random variable is used to represent the amount of produced/consumed resource. By Maximum A Posterior (MAP) estimation, the model is trained by the system logs. After that we estimate the probability, for each resource, that the producing satisfying the consuming. In this way, we evaluate whether this service can reuse the resource inside. At last, we calculate the profitability by computing the expectation of difference between consuming and producing on funds, which is a special resource.

This paper is organized in following way: Section 2 studies our solution framework. Section 3 discusses the detailed techniques in each step. The experiment is reported in Section 4. Section 5 takes a review on literature. Section 6 concludes our work.

II. Framework

The solution framework, as illustrated in Fig. 1 is introduced. This framework consists of two phases:

- Phase 1: An **Element Extractor** derives the resource list and activity list from the service description. Then the **Log Estimator** learns the producing and consuming matrix from the system log.
- Phase 2: A **Pattern Slicer** cuts the service pattern by variant resources. After that the **Resource Evaluator** estimates the satisfaction of each resource and evaluate the profitability.

In this section, we mainly discuss the basic concepts and notations with the help of some examples. Both four basic ingredients in our framework, namely **Element Extractor**, **Log Estimator**, **Pattern Slicer** and **Resource Evaluator** will be studied one by one in section 3.

A. Concepts and Notations

In this section, we review some basic concepts in service. The first concept is the activity in service.

Definition II.1 (Activity)**.** *An activity, denoted as a, is a basic and atomic function component in service.*

We use $\mathcal{A}$ to denote the set of activity and iterator a_i to represent the i'th activity in $\mathcal{A}$. The size of $\mathcal{A}$ is presented as $|\mathcal{A}|$. In the research on web service, an activity can be either an atomic service or a composed service [7]. In business process management, (BPM), it is called activity [8]. For general discussion, we just call it activity, without specify whether a web service or an activity in BPM.

The basic attribute of an activity a, is its functional description. It can be WSDL [9], OWL-S [10] in Web service and natural language description in BPMN [11]. Both these formats are insufficient and inconvenient in mining the resource exchange.

Example II.1. *Uploading video is a common activity in Youtube. In this activity, a customer selects a local video, which is either edited or shot by himself/herself, and uploaded to Youtube website. From the perspective of resource exchange, it spends the CPU time, network traffic and disk storage to exchange a valuable resource, namely the digital video.*

This example shows that the natural language description is long and inaccurate for our evaluation on how good is this activity for the entire service. Therefore we apply the SDPL model [12], our previous work, which systemic defines the service and its activities by the resource/data exchange.

Definition II.2 (Resource)**.** *A resource, denoted as r, is a kind of valuable things, which can be consumed and created repeatedly.*

Note that the resource can be either visible, e.g. a book and a machine, or intangible, e.g. the valuable brand. Funds, or money, is a special and essential resource for service. It is so important in the evaluation because all activities are designed to maximize the earning funds in service. To specify funds with other resource, we denote it as r_1, while other resources are denoted as r_i ($i > 1$). Besides, we use $\mathcal{R}$ to represent the set of resources involved in service and $|\mathcal{R}|$ to denote the total number of resources. Note that different resources are under different measurement. For example, the CPU time is counted by *Second*, and disk storage is counted by *GigaByte*. In our discussion, we ignore the measurement and just study the number relation for simplification.

In SPDL, an activity's functional description is represented as the resource operation list. Here we mainly discuss two basic resource operations:

- **Consuming** means using up an resource in an activity.
- **Producing** means creating a new resource instance in an activity.

To quantitatively discuss the resource amount in practice, we introduce the **producing matrix** O, and **consuming matrix** C. Both in shape of $|\mathcal{R}| \times |\mathcal{A}|$. Each row represents

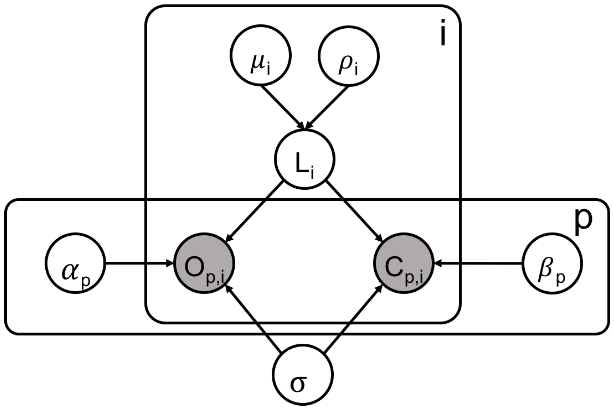


Figure 2: The probability model on resource operation

a resource, and each column represents an activity. The cell $o_{p,i}$, namely cell at the p'th row and the i'th column, means activity a_i producing $o_{p,i}$ resource p. Note that the amount of resource, either consuming or producing, vary in different activity instance. In other words, among the instances of uploading video, one may use 500MB in disk storage and another may use 1GB. Therefore, both O and C is a random variable instead of the constant value. To distinguish the random variable with the normal value, we use the bold style on the notations, namely $\mathbf{O}$ and $\mathbf{C}$, for the random variable.

Example II.2. *In previous example, uploading video involves four resources: CPU time, network traffic, disk storage and video. We can mark them as r_2, r_3, r_4 and r_5. Besides, r_1 is also declared for funds. The former three resources, r_2, r_3 and r_4, are consumed while the last resource, namely video r_5, is created in this activity. Formula* (1) *shows the consuming matrix and producing matrix means using 1800 seconds CPU time, 150 network traffic and 0.5 GigaByte storage, and getting a video instance.*

$$\mathbf{C} = \begin{pmatrix} 0, 1800, 150, 0.5, 0 \end{pmatrix} \mathbf{O} = \begin{pmatrix} 0, 0, 0, 0, 1 \end{pmatrix} \tag{1}$$

Till now, we can use resource consuming matrix $\mathbf{C}$ and producing matrix $\mathbf{O}$ to represent the resource operation. The detailed calculation from log will be discussed in following section.

B. Service Pattern

Before introducing the service pattern, we study a probability model which is feasible to learn the numeral relation between producing matrix and consuming matrix. Consider following generative probability model:

- For each activity a_i:
- A multiplier L_i is sampled from a normal distribution $L_i \sim N(\mu_i, \rho_i)$, in which μ_i and ρ_i are two parameters.
- For each resource r_p:
- The producing amount $\mathbf{O}_{i,p}$ is sampled from the normal distribution $\mathbf{O}_{i,p} \sim N(\alpha_p * L_i, \sigma)$ in which α_p is the basic producing rate parameter and σ is a parameter.
- The consuming amount $\mathbf{C}_{i,p}$ is sampled from the normal distribution $\mathbf{C}_{i,p} \sim N(\beta_p * L_i, \sigma)$ in which β_p is the basic consuming rate parameter.

In this probability model, both $\mathbf{O}$ and $\mathbf{C}$ are observation, L is a latent variable whose domain is $(0, 1)$, α,β,μ,ρ and σ are parameters. We simply denote $\mathbf{\Theta} = \{\alpha, \beta, \mu, \rho, \sigma\}$ as a collection for all parameters in this model. By introducing this generative model, we can estimate the joint conditional probability $P(\mathbf{O}_{i,p}|\mathbf{C}_{i,q}, \mathbf{\Theta})$ which means the likelihood of $\mathbf{O}_{i,p}$ resource p is produced when $\mathbf{C}_{i,q}$ is used in activity i. The expectation of resource producing can also be estimate by $\mathbb{E}(\mathbf{O}_{i,p}|\mathbf{C}_{i,q}) = \int_{\mathbf{O}_{i,p}} \mathbf{O}_{i,p} P(\mathbf{O}_{i,p}|\mathbf{C}_{i,q}, \mathbf{\Theta})$

Definition II.3 (Service Process)**.** *a service process, p is a tuple $< \mathcal{A}, \mathcal{R}, \mathbf{O}, \mathbf{C} >$, where $\mathcal{A}$ is the activity set, $\mathcal{R}$ the resource set, $\mathbf{O}$ the resource producing matrix and $\mathbf{C}$ the resource consuming matrix.*

This definition, integrating the resources, activities and their operations together, provides a formate and structure for evaluation. Its intuition is to compare one service company with its competitor in profitability, the manager needs to summarize the activities, resources and resource exchanging rates. For an activity i and resource p, the competitor performs better if it can get a higher $\mathbf{O}_{i,p}$ given the same $\mathbf{C}_i$.

Example II.3. *Two companies, marked as $P1$ and $P2$, both provide disk renting service. There is only one activity and two resources, funds and disk storage, involved for both $P1$ and $P2$. Formula* (2) *shows the consuming matrix and producing matrix of company $P1$.*

$$\mathbf{C} = \begin{pmatrix} 0, 10 \end{pmatrix} \mathbf{O} = \begin{pmatrix} 20, 0 \end{pmatrix} \tag{2}$$

Similar, Formula (3) *illustrates the consuming and producing matrix for company $P2$.*

$$\mathbf{C} = \begin{pmatrix} 0, 10 \end{pmatrix} \mathbf{O} = \begin{pmatrix} 40, 0 \end{pmatrix} \tag{3}$$

In comparing with $P2$, $P1$ uses 10 units disk storage and gets only 20 units funds, while $P2$ can get 40 unit funds. In other words, $P2$ has a higher resource exchanging rate comparing with $P1$.

This example emphasize that resource consuming matrix and producing matrix are two important features in evaluating the profitability.

we have reviewed the required definition on service process and the generative probability model. The service pattern $\mathcal{P}$, although may be defined differently in literature [13][14][8], in this paper is represented by the trained parameters $\mathbf{\Theta}$ in the generative probability model. The parameters, which are initialized randomly, are called trained if the model has been fitted with the log records by the **Log Estimator**.

III. Framework Ingredients

With the definition and formalization on activity, resource and service pattern, we can study the ingredients in detail, see Fig. 1.

A. Element Extractor

Element Extractor takes the service description as input and summarizes the activity list and resource list for evaluation. Note that the raw service description is semi-structured. In WSDL, OWL-S and BPMN, the process and interface is well formulated while the function part is presented in natural language. In our previous work, a general resource-oriented service description language is introduced, called **Service Pattern Description Language (SPDL)** [12], which can provide a new way to organize the activity and the resource operation. With its help, the resource operations and activities are bind in the SPDL formate file. Also, for the raw service description formate, e.g. BPMN, we provide an extendable convertor in [14]. Therefore this part is omit here due to page limit.

B. Log Estimator

Table I presents an example of some log fragments. It involves the activity, the amount of resource, consumed and produced, and the timestamp information. For example, the first record shows that at 13:40 May 1st, 2016, a user uploaded a video using 1800 CPU, 150 Network Traffic and 0.5 Disk. In other three records, the resource consuming amount is different with the first one while the producing amount is the same. Therefore, we can not judge the efficiency of resource exchanging by one record. Instead, the whole records should be taken into consideration. A straight forward solution is averaging them as the expectation on resource usage/production. However, it is rough and poor-performed when the variance is too large. Therefore, we learn an **Log Estimator** by the generative probability model mentioned in section II-B. It takes the log, activity list and resource list as input and learn the generative model for consuming matrix $\mathbf{C}$ and producing matrix $\mathbf{O}$. The intuitive understand is that, since the consuming matrix and producing matrix are tightly relevant to the observations, which can be covered by the log, see Fig. 2. We'd like to estimate the joint probability given the observations by Maximum A Posterior.

$$\begin{aligned}
&P(\mathbf{\Theta}|\mathbf{C},\mathbf{O})\\
=&\prod_{t=1}^{T} P(\mathbf{\Theta}|\mathbf{C}^{(t)},\mathbf{O}^{(t)})\\
\propto&\prod_{t=1}^{T}\int_0^1 P(\mathbf{O}_{p,i}^{(t)}|\alpha_p,\sigma,L_i)P(\mathbf{C}_{p,i}^{(t)}|\beta_p,\sigma,L_i)P(L_i|\mu_i,\rho_i)dL_i\\
=&\prod_{t=1}^{T}\Psi(\mathbf{O}_{p,i}^{(t)},\mathbf{C}_{p,i}^{(t)};\mathbf{\Theta})
\end{aligned} \tag{4}$$

where the upper script t denotes the t'th record in log. And Ψ is defined as follows

$$\Psi(\mathbf{O},\mathbf{C}) = \int_L P(\mathbf{O}|L,\mathbf{\Theta})P(\mathbf{C}|L,\mathbf{\Theta})P(L|\Theta)dL \tag{5}$$

Given the observations $\mathbf{C}$ and $\mathbf{O}$ in log record, Formula (4) can be maximized by optimize the parameter $\mathbf{\Theta}$ via Stochastic Gradient Decent (SGD). The loss function, see formula (6), summarizes the posterior probability and regular term.

$$\mathcal{L} = -\sum_{t=1}^{T}\log\Psi(\mathbf{O}_{p,i}^{(t)},\mathbf{C}_{p,i}^{(t)};\mathbf{\Theta}) + \lambda\|\mathbf{\Theta}\|_2 \tag{6}$$

In this procedure, the model and parameters are trained for better fitting the observations. The model with trained parameters $\mathbf{\Theta}$ is the service pattern $\mathcal{P}$ for given log record.

C. Pattern Slicer

We witness that there are plenties of resources, $\mathcal{R}$ involved in a service pattern $\mathcal{P}$. For better evaluating the service pattern, the pattern slicer is introduced to specify the life cycle of each resource. The slicer takes the service pattern $\mathcal{P}$ as input and outputs the resource patterns $\{\mathcal{P}_1,\mathcal{P}_2,...,\mathcal{P}_{|R|}\}$. Consider a resource r_p, namely the p'th resource, the partial parameter α_p, β_p, σ, ρ, μ are involved. Recall that in the trained model, α_p and β_p are both fixed variable in $\mathbb{R}$. Parameter σ, which is also in $\mathbb{R}$ is shared for all resources. Both ρ and μ are vectors in length of $|\mathcal{A}|$ and they are shared for all resources. Therefore, the resource pattern can be represented as $\mathcal{P}_i = \{\alpha_i,\beta_i,\sigma,\rho,\mu\}$.

D. Resource Evaluator

To judge how good is the resource pattern, we'd like to estimate the probability that the resource can be well-recycled. Specifically, if the amount of resource consumed in the service process is larger than the resource amount produced, the service process can not be sustained. Formally speaking, for a resource pattern $\mathcal{P}_i = \{\alpha_i,\beta_i,\sigma,\rho,\mu\}$, the i'th resource is considered. This resource can be recycled if and only if $\sum_{j=1}^{|\mathcal{A}|}\mathbb{E}[\mathbf{O}_{j,i}] \geq \sum_{j=1}^{|\mathcal{A}|}\mathbb{E}[\mathbf{C}_{j,i}]$. In other words, we can estimate the expectation of rest resource amount and

Table I: Log fragment example

Activity	Resource 1 (Funds) consumed /produced	Resource 2 (CPU) consumed /produced	Resource 3 (Network) consumed /produced	Resource 4 (Disk) consumed /produced	Resource 5 (Video) consumed /produced	Start time	End time
a(upload)	0/0	1800/0	150/0	0.5/0	0/1	2016/5/1 13:40	2016/5/1 13:50
a(upload)	0/0	2000/0	200/0	0.8/0	0/1	2016/5/1 14:01	2016/5/1 14:03
a(upload)	0/0	1600/0	180/0	0.6/0	0/1	2016/5/1 14:10	2016/5/1 14:18
a(upload)	0/0	1830/0	160/0	1.2/0	0/1	2016/5/1 14:20	2016/5/1 14:26

check that if it is equal or larger than 0.

$$\mathbb{E}[\sum_{j=1}^{|\mathcal{A}|}(\mathbf{O}_{ij}-\mathbf{C}_{ij})] = \int_{\mathbf{O}}\int_{\mathbf{C}}(\sum_{j=1}^{|\mathcal{A}|}(\mathbf{O}_{ij}-\mathbf{C}_{ij}))\Psi(\mathbf{O},\mathbf{C})d\mathbf{O}d\mathbf{C} \tag{7}$$

The probability that the resource can be well-recycled, called **well-recycled probability**, can be calculated by formula (8).

$$P(\sum_{j=1}^{|\mathcal{A}|}(\mathbf{O}_{ij}-\mathbf{C}_{ij}) > 0|\mathcal{P}_i) = \int_{\sum_{j=1}^{|\mathcal{A}|}(\mathbf{O}_{ij}-\mathbf{C}_{ij})>0}\Psi(\mathbf{O},\mathbf{C})d\mathbf{O}d\mathbf{C} \tag{8}$$

Furthermore, we can summarize all resources on their **well-recycled probability** as the **Resource Sustainability Probability** (RSP), which can be estimated by (9)

$$\prod_{i}^{|\mathcal{R}|}P(\sum_{j=1}^{|\mathcal{A}|}(\mathbf{O}_{ij}-\mathbf{C}_{ij}) > 0|\mathcal{P}_i) \tag{9}$$

Therefore, with the resource pattern learned before and Eq. (7), the service manager can estimate whether the service process can work well on resource usage in the future according to its past log records. It can generate a resource utilization report which contains the expectation of rest amount on various resource.

Another important perspective of service pattern evaluation is to judge the profitability. Recall that we mentioned a special resource, namely the funds, marked as r_1, which determines the profitability of service pattern. In our framework, we can estimate the **earning ratio** as the indicator for profitability. The earning ratio, or called yields and returns, is always defined as the ratio of earning on cost. In our framework, the earning is represented as the differ of produced funds amount with the consumed, namely $\mathbf{O}_1 - \mathbf{C}_1$. The subscript 1 means the first resource, namely the funds r_1. Then, the earning ratio can be calculate as the expectation on $(\sum_{j=1}^{|\mathcal{A}|}(\mathbf{O}_{1j}-\mathbf{C}_{1j}))/\sum_{j=1}^{|\mathcal{A}|}\mathbf{C}_{1j}$.

$$\begin{aligned}&\mathbb{E}[\frac{\sum_{j=1}^{|\mathcal{A}|}(\mathbf{O}_{1j}-\mathbf{C}_{1j}))}{\sum_{j=1}^{|\mathcal{A}|}\mathbf{C}_{1j}}] \\ &= \int_{\mathbf{O}}\int_{\mathbf{C}}\frac{\sum_{j=1}^{|\mathcal{A}|}(\mathbf{O}_{1j}-\mathbf{C}_{1j}))}{\sum_{j=1}^{|\mathcal{A}|}\mathbf{C}_{1j}}\Psi(\mathbf{O},\mathbf{C})d\mathbf{O}d\mathbf{C}\end{aligned} \tag{10}$$

With formula (10) and the service pattern $\mathcal{P}$ estimated from log records, we can calculate the earning ratio and provide a profitability report for the service process manager.

IV. Experiments

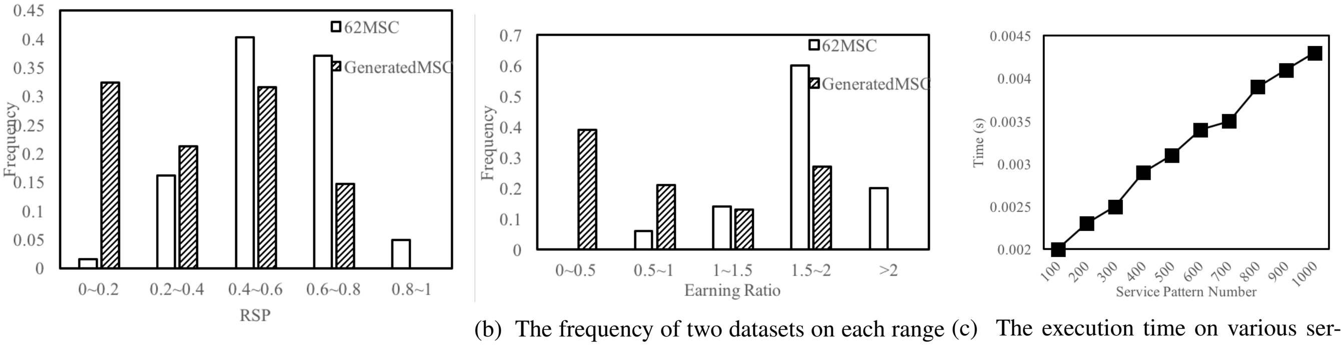
In this section, we report the experiment results of our evaluation framework on both efficiency and effectiveness. Both experiments are implemented on Matlab R2016A. The execution environment is a MacBook Pro with Intel i7, 3.1GHz, 16G Memory.

A. Dataset

The dataset used in our experiment is the combination of real service process and synthetic log records. The basic processes are extracted from the 62 enterprises from 355 public Modern Service Companies, called *62MSC*, in the growth enterprise market, the second-board market which is very similar to the NASDAQ Stock Market, in China [15]. Table II shows the statics on the 62 basic processes. To generate more processes for experiments, we randomly exchange some activities between each two processes. This dataset contains 1000 process, called *GeneratedMSC*, whose statistic is similar to the original *62MSC*, see Tab. II. Another data required in service pattern evaluation is the log records on resource usage, which is usually the business secret for most companies. It is hard for us to collect such data but possible for practical process manager on evaluating owing process. For example, for Youtube, its managers can access its resource usage both CPU, Network and disks for evaluating Youtube's service pattern. Therefore, we generate a collection of resource log for studying the effectiveness and efficiency of our approach. Each record is generated by randomly selecting a resource to use/produce by a uniform distribution and the resource amount is sampled from a normal distribution whose expectation is the a constant relevant to the activity.

Table II: Statistic on Dataset

Datasets	# Process	# Activity Per Process (Avg, Min, Max)	# Resource Per Process (Avg, Min, Max)
62MSC	62	24.24, 5, 43	6.01, 3, 10
GeneratedMSC	1000	24.11, 5, 50	5.74, 3, 12



(a) The frequency of two datasets on each range of RSP

(b) The frequency of two datasets on each range of Earning Ratio

(c) The execution time on various service pattern number

Figure 3: The experiment results

B. *Effectiveness*

In this first experiment, we apply our evaluating approach on both *62MSC* and *GeneratedMSC*. Note that on one hand, the 62 companies in *62MSC* public offering on second-board market which guarantees their profitability. In other words, their service patterns have been well-studied by most investment institutions. On the other hand, our generated service patterns in *GeneratedMSC* are randomly composed. Their profitability and resource recycling are lack of practical base. Figure 3a shows the difference about the distribution of *62MSC* and *GeneratedMSC* on **Resource Sustainability Probability** (RSP). The x-axis contains the 5 ranges of RSP, namely $0 \sim 0.2$, $0.2 \sim 0.4$, $0.4 \sim 0.6$, $0.6 \sim 0.8$, $0.8 \sim 1$. The y-axis indicates the frequency of service processes on each range. We can see that the *62MSC*, namely the real service pattern set, has a higher frequency on 0.4 0.8, comparing with *GeneratedMSC*. Recall that the higher RSP means the service pattern is more sustained on resource recycling. In the comparison on these two datasets, we can get the conclusion that the service patterns in *62MSC* has a higher stability on resource recycling. Just as we mentioned before, the service patterns in *62MSC* have been well studied by plenties of investment institutions. The resource recycling provides a solid foundation on its profits.

Another important indicator is the **earning ratio**, which shows the profitability of service pattern. Similar to RSP, we set 5 ranges of earning ratio, namely, $0 \sim 0.5$, $0.5 \sim 1$, $1 \sim 1.5$, $1.5 \sim 2$ and larger than 2. The result is illustrated in Figure 3b. We can see that about 60% service patterns in *62MSC* have the earning ratio from 1.5 2. While in *GeneratedMSC*, near 40% service patterns have the earning ratio lower than 0.5. The reason is similar to RSP, the real service patterns in *62MSC* are maturing and widely used in practice. Both these two experiments show the effectiveness of our service pattern evaluating approach.

C. *Efficiency*

This experiment, we report the scalability of our approach on the number of process. In this experiment, we mix the *62MSC* and *GeneratedMSC* together. We conduct this experiment by randomly sampled 100, 200, 300, 400, 500, 600, 700, 800, 900 and 1000 service patterns in the mix 1062 patterns. For each sampled service patterns, we train our model on 70% samples by Stochastic Gradient Descent on the loss function and get the trained parameters. Then we apply our trained model on the rest 30% samples and record the executing time. Note that the configuration and settings are kept the same for all groups. The result is reported in Figure 3c, which shows that our approach is linear time complexity, namely $O(|\mathcal{P}|)$ where $|\mathcal{P}|$ is the number of processes.

V. Related Works

In this section, we take a quick review on the literature on the service pattern and some related works on service evaluation. The study on service evaluation, in the past decade, majorly concentrates on the **Quality of Service**(QoS) [4]. QoS provides an analyzing framework on services, including QoS prediction and QoS based applications.

The former is the problem to estimate the QoS of composed service, containing several atomic services, whose QoS are given. [16] emphasized the usage of QoS prediction in Cloud Services. [17] [18] used QoS prediction to improve the efficiency in warehouse scale computers. For different QoS indicators, the prediction approaches are different. The key challenge comes from the service process control flow, which determines the sequence of atomic services. Besides basic sequential flow, there are parallels and branches [4] [19]. Therefore the prediction faces the challenge because for different control flow may have different way to summarize the QoS. QoS prediction provides a basic way to understand the evaluation on service pattern. For example, [5] used neighborhood integrated matrix factorization in

QoS prediction From the perspective of QoS prediction, the resource producing matrix and consuming matrix are both the QoS indicators which can be predicted, just as we do in section 3. However, classical QoS prediction cannot directly apply on our problem, because the resource producing and consuming is not related to the control flow while tightly related to the basic exchanging ratio in the atomic services, we called activities.

QoS based applications, such as QoS based service selection and composition. [20] introduced the personalized QoS-based service recommendation. [21] studied the QoS to evaluate the trustworthiness of cloud service. [22] discussed the importance of QoS on heterogeneous datacenters Both these applications use QoS as a foundation to support complex decision. They provide us a great reference to apply service pattern evaluation in aiding decision making. In our previous work [12], we have studied the application of service pattern on service designing.

Besides QoS issues, there are some interesting researches on service pattern and value creation. In [8], workflow patterns are studied, whose basic components are activities and control flow, in which the resource exchange is not studied. E3 value [23], is a famous framework support value analyzing in businesses. It provides an extendable visualization tool helping process manager to draw their process out and observe the value flow between each activity. However, comparing with our framework, E3 value requires too many manual works and lacks automatic quantitatively analyzing tool.

VI. Conclusion

We witness the inconvenient of service owners and managers in analyzing their service process, the procedure requires so many boring and repeat statistical works. Therefore we introduce a novel and extendable framework which takes the service design and system logs to extract the service pattern. After that the service pattern can be evaluated for each resource on its stability on resource recycling. The report on resource recycling probability and earning ratio are provided as output. This framework contains a generative probability model to estimate the resource producing/consuming in each activity. This model can be used in predicting the resource using by trained on the log records. We collect the data from the second-board market and conduct the experiments to report the effectiveness and efficiency of our model.

Acknowledgment

This work was supported by Zhejiang Provincial Natural Science Foundation of China under grant No.LY15F02007.

References

[1] S. Wang and H. Mao, "The strategy research of modern service industry development," *Asian Social Science*, vol. 8, no. 2, p. 185, 2012.

[2] R. Kalakota and M. Robinson, *e-Business*. Addison Wesley Roadmap for Success, 1999.

[3] J. Kim, Y. Park, C. Kim, and H. Lee, "Mobile application service networks: Apple's app store," *Service Business*, vol. 8, no. 1, pp. 1–27, 2014.

[4] L. Zeng, B. Benatallah, A. H. Ngu, M. Dumas, J. Kalagnanam, and H. Chang, "Qos-aware middleware for web services composition," *IEEE Transactions on software engineering*, vol. 30, no. 5, pp. 311–327, 2004.

[5] Z. Zheng, H. Ma, M. R. Lyu, and I. King, "Collaborative web service qos prediction via neighborhood integrated matrix factorization," *IEEE Transactions on Services Computing*, vol. 6, no. 3, pp. 289–299, 2013.

[6] S. Ding, X.-J. Ma, and S.-L. Yang, "A software trustworthiness evaluation model using objective weight based evidential reasoning approach," *Knowledge and information systems*, vol. 33, no. 1, pp. 171–189, 2012.

[7] J. Rao and X. Su, "A survey of automated web service composition methods," in *International Workshop on Semantic Web Services and Web Process Composition*. Springer, 2004, pp. 43–54.

[8] W. M. Van Der Aalst, A. H. Ter Hofstede, B. Kiepuszewski, and A. P. Barros, "Workflow patterns," *Distributed and parallel databases*, vol. 14, no. 1, pp. 5–51, 2003.

[9] E. Christensen, F. Curbera, G. Meredith, and S. Weerawarana, "Web services description language (wsdl) 1.1," 2001, wSDL.

[10] OWL-S, "http://www.daml.org/services/owl-s/."

[11] S. A. White, "Process modeling notations and workflow patterns," *Workflow Handbook*, vol. 2004, pp. 265–294, 2004, bPMN.

[12] J. Yin, Z. Luo, Y. Li, and Z. Wu, "Service pattern: An integrated business process model for modern service industry," *IEEE Transactions on Services Computing*, 2016.

[13] J. Yin, Z. Luo, Y. Li, B. Fan, W. Liu, and Z. Wu, "Towards a service pattern model supporting quantitative economic analysis," pp. 95 – 102, 2014 2014.

[14] Y. Li, Z. Luo, J. Yin, L. Xu, Y. Yin, and Z. Wu, "Enterprise pattern: integrating the business process into a unified enterprise model of modern service company," *Enterprise Information Systems*, vol. 11, no. 1, pp. 37–57, 2017.

[15] M. M. Yao, D. Wu, X. B. Wu, Z. H. Wu, and S. C. Chen, "Business model innovation of modern service company: A value network perspective," in *Management of Technology (ISMOT), 2012 International Symposium on*, ser. Management of Technology (ISMOT), 2012 International Symposium on. IEEE, 2012, pp. 76–80.

[16] Z. Zheng, X. Wu, Y. Zhang, M. R. Lyu, and J. Wang, "Qos ranking prediction for cloud services," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 6, pp. 1213–1222, 2013.

[17] Y. Zhang, M. A. Laurenzano, J. Mars, and L. Tang, "Smite: Precise qos prediction on real-system smt processors to improve utilization in warehouse scale computers," in *Proceedings of the 47th Annual IEEE/ACM International Symposium on Microarchitecture*. IEEE Computer Society, 2014, pp. 406–418.

[18] H. Yang, A. Breslow, J. Mars, and L. Tang, "Bubble-flux: Precise online qos management for increased utilization in warehouse scale computers," in *ACM SIGARCH Computer Architecture News*, vol. 41, no. 3. ACM, 2013, pp. 607–618.

[19] Q. Yu, "Qos-aware service selection via collaborative qos evaluation," *World Wide Web*, vol. 17, no. 1, pp. 33–57, 2014.

[20] X. Chen, Z. Zheng, X. Liu, Z. Huang, and H. Sun, "Personalized qos-aware web service recommendation and visualization," *IEEE Transactions on Services Computing*, vol. 6, no. 1, pp. 35–47, 2013.

[21] S. Ding, S. Yang, Y. Zhang, C. Liang, and C. Xia, "Combining qos prediction and customer satisfaction estimation to solve cloud service trustworthiness evaluation problems," *Knowledge-Based Systems*, vol. 56, pp. 216–225, 2014.

[22] C. Delimitrou and C. Kozyrakis, "Paragon: Qos-aware scheduling for heterogeneous datacenters," in *ACM SIGPLAN Notices*, vol. 48, no. 4. ACM, 2013, pp. 77–88.

[23] Y. S. Kim, C. K. Cho, Y. D. Ko, and H. Jee, "E3 value concept for a new design paradigm," in *DS 68-2: Proceedings of the 18th International Conference on Engineering Design (ICED 11), Impacting Society through Engineering Design, Vol. 2: Design Theory and Research Methodology, Lyngby/Copenhagen, Denmark, 15.-19.08. 2011*, 2011.