ICSP'04 Proceedings

Evaluation of Action Prediction Method Using Inductive Learning with N-gram

JIN'AN XU† TOSHIHIKO ITOH† KENJI ARAKI† KOJI TOCHINAI‡
†Graduate School of Engineering, Hokkaido University Kita13, Nishi 8, Kita-ku, Sapporo, 060-8628, JAPAN
Phone&Fax:+81-11-706-7389 E-mail:{xja, t-itoh, araki}@media.eng.hokudai.ac.jp
‡Graduate School of Business Administration, Hokkai-Gakuen University
Asahimachi 4-1-40, Toyohira-ku, Sapporo, 062-8605 JAPAN
Phone:+81-11-841-1161 ext.634 Fax:+81-11-824-7729 E-mail: tochinai@econ.hokkai-s-u.ac.jp

Abstract：*Being society aging, an intelligent room is needed for the aged or handicapped. The important ingredient of such a system is how to predict the next action. In this paper we describe how to solve the problem of predicting inhabitant action in an intelligent room that we called learning room. We have proposed a method to predict user action using Inductive Learning (IL) with N-gram. The system based on our proposed method is able to acquire the immanent causality rules automatically from data pairs by means of IL. Since our system unified IL and N-gram, it demonstrates good accuracy for the simulated data. The system showed high dynamic adaptive capability.*

Key words: *Inductive Learning, N-gram, Action prediction, Dynamic adaptation*

1. Introduction

Our goal is to develop an action prediction system that can be dynamically adapted to each user. If future intention or action of a user can be correctly predicted, it means that the system can give the user much convenience. Such kind of intelligent systems need to have high prediction accuracy, high dynamic adaptation capability and high flexibility.

To realize an intelligent system like the one described above, one of the most difficult problem to face is how to predict user actions. Since each user behaves differently, it's a difficult task to collect action history for each user.

A possible solution to this problem is to use the Rule-based Machine Learning method [1] as analytical approach, furnish the system with rules in advance. However, to prepare these rules requires huge labor since habits and preferences vary from one user to another, it is impossible to prepare rule dictionary in order to deal with each user.

On the other hand, most traditional prediction techniques use statistical approaches, which are classified into Point-Based Prediction system [2], and Pass-Based Prediction system [3]. Almost all Point-Based Prediction systems use the Markov model or the Expectation Maximization Algorithm. However, the main problem of these systems is that these require huge database that needs to be prepare by hand. On the other hand, many researchers have proposed methods that use N-gram to analyze time-series data as the Pass-Based Prediction System. In this approach, the prediction accuracy depends on the amount of training data. Prediction becomes impossible when the predicted target's appearance frequency is low. This situation is known as the "Data Sparseness Problem". In order to solve this problem, Example-based learning techniques have been applied to the action prediction system [4]. Such a system can increase prediction accuracy if given enough training data. However, system cannot predict for each user's action because it is not easy to collect sufficient data for each user.

K.Araki and K. Tochinai have proposed a natural language processing method using Inductive Learning (IL) [5]. With this approach, immanent rules are acquired through recursive extraction of the common part and different parts from data pairs [6]. The effectiveness of this method has already been proven in natural language processing [7,8,9].

With this technique rules can be acquired from a single appearance of performed action history automatically. Since the acquired rules would express the user's habits and preferences, the system could have dynamic adaptive capability for each user. Therefore, we adopt the N-gram into our system to find regularities in user's actions. As a result, we unified IL and N-gram as a Point-Pass-Based action prediction method.

In this paper, we outline our proposed method , discuss the optimization of the experimental results, and indicate potential directions of future research.

2. Inductive Learning

We used an IL technique proposed by K. Araki and K. Tochinai [4]; here we will introduce its basic idea [4]. Table 1 shows an example of extraction of correspondences from unknown character strings prepared by human being.

Table 1. Example of Inductive Learning

Input1	α θ σ φ δ λ υ	
Input2	Ξ Σ φ δ γ Φ Θ	
Segment1	α θ σ	Ξ Σ
Segment2	φ δ	φ δ
Segment3	λ υ	γ Φ Θ

According to this heuristics, we can extract different parts and common parts as segments from data pairs. Moreover primitives can be extracted from the segments with this idea too. Table 2 shows an example of the primitives extracted from segments. The idea of Inductive Learning is to automatically acquire knowledge in multistage steps by extracting common parts and different parts. The learning process of our system is based on these processes as mentioned above.

Table 2: Example of recursively extraction

Segment1	α θ σ	γ Φ Θ
Segment2	θ σ γ μ	Φ Θ Σ
Primitive1	α	γ
Primitive2	θ σ	Φ Θ
Primitive3	γ μ	Σ

3. Fundamental Concept

We aim is to explore the heuristics for developing a general-purpose action prediction system that could adapt dynamically to each user. We assume our system is an intelligent room system (Learning Room). We would like to investigate the engineering principles behind it too. The prediction problem is even more difficult for a "Learning Room" where multiple inhabitants perform multiple actions at the same or different times. As a first step, we investigate a portion of this problem: predicting a single user's actions and intentions.

The Learning-Room consists of eight artificial elements, a door (D), TV, TV program (P), curtain(C), light (L), air-conditioner (AC), telephone (TEL), and alarm clock (Al). We consider some natural elements like time, temperature and brightness. Artificial element, we mean elements whose state can be changed by the user, while the natural elements cannot be changed by the user's actions. Each one of these elements has some fixed values (see Table3).

Table3: Element's Fixed Value

Element	Fixed value
Day of week	Monday…Sunday
Time	hh , mm
Brightness	50,100…1450,1500(Lux)
Temperature	16,17…34,35
Door	Doff, Don
TV	Toff, 1ch, 2ch…60ch
TV-program	None, Sports, News…Film
Curtain	Coff, Con
Light	Loff, Lon
Air conditioner	Aoff, 16, 17, 18…35
Telephone	Teoff, Ring, Teon
Alarm clock	Aloff, Alon

The user's actions need to be accessible in some tangible form, and then we can generate a sequence of historic actions as a string of action symbols or we can represent individual actions as separate states with corresponding state transitions.

We symbolize room state (S_k^n) as follows:

$$S_k^n = (N_k^n , A_k^n) \quad (1)$$

N_k^n expresses natural elements vector space, and A_k^n means artificial element vector space as follows:

$$N_k^n = (W_k^n , H_k^n , M_k^n , B_k^n , T_k^n) \quad (2)$$

$$A_k^n = (D_k^n , TV_k^n , P_k^n , C_k^n , L_k^n , AC_k^n , TEL_k^n , Al_k^n) \quad (3)$$

, where, W, H, M, B, T express: day of week, hour, minute, brightness and temperature respectively. D, TV, P, C, L, AC, TEL, AL express door, TV, TV program, curtain, light, air conditioner, telephone and alarm clock respectively. K expresses the date of data, n express the data number.

4. Using Inductive Learning and N-gram

According to the basic idea of Inductive Learning, by extracting the different part from artificial elements states, we can acquire initial rules as follows:

$$Rule_k^n = (S_k^n , Action_k^n) \quad (4)$$

, where, $Action_k^n$ expresses the actions performed by user. For examples:

S_1^1 = (Monday, 7:31,750,24,Doff, 3ch,News, Con, Loff, Aoff, **Ring,** Aloff)

S_1^2 = (Monday, 7:31,750,24,Doff, 3ch,News, Con, Loff, Aoff, **Teon**, Aloff)

S_1^3 = (Monday, 7:31,750,24,Doff, **Toff, None**, Con, Loff, Aoff, Teon, Aloff)

Then we can acquire two initial rules, such as:

$Rule_1^1 = (S_1^1$, **Teon**) and $Rule_1^2 = (S_1^2$, **Toff&None**).

In order to acquire rule's adaptation capability, we perform abstraction process for initial rules using IL. The abstraction process algorithm is as Figure 1.

```
Algorithm abstraction process (A: user's current action; n:
acquired initial rule's number of action A, m: element's number)

  Begin
   For i = 0 to n-1
      For j = 0 to m-1
        If element's states are different, using a variable @j as
        the element's value to generate the abstracted rule.
             Else taking the value into the abstracted rule
        End If
        End For
        End For
        Register the abstracted rules into rule dictionary
        Else Return ("No abstraction")
  END
```

Figure 1 Rule Abstraction Process

Using Inductive Learning, we can extract user's action sequence as a script. For example, from the data set of S_1^1, S_1^2, S_1^3; we can extract this script: (Ring ,Teon ,Toff&None). In this paper , we use bigram to analyze action's conditional probability.

5. Outline

This section describes the procedure of our proposed method. As shown in Figure 2, our system procedure consists of prediction process, proofreading process, feedback process and learning process.

In Prediction Process, as we give certain amount of dataset to the system, it makes prediction by pattern matching, while referring to the data history and the rule dictionary. With reference to the rule dictionary, the highest credibility evaluated rule is selected by the system. With reference to the data history, the room state is acquired in accordance with the user action.

In proofreading process, when the predicted result matches the user's action, the state will be registered into the state history. Otherwise, the user performs proofreading, i.e., the user judges whether the action is correct or not. If not, the corrected result will be registered into the proofreading history, and then feedback process takes place.

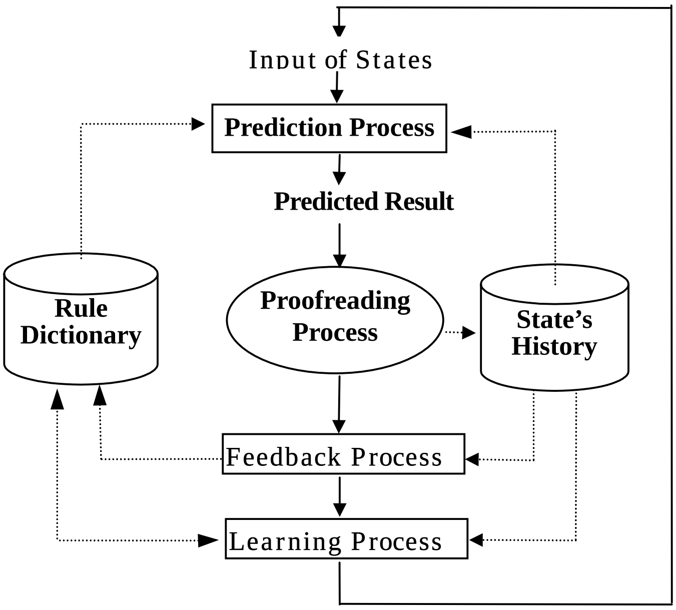

Figure 2: Procedure

In feedback process, rules credibility is updated based on four values tracked for each rules those values are correct prediction frequency (CF), error prediction frequency (EF), correct prediction rate (CR) and error prediction rate (ER). The system automatically determines the superior of several competing rules. Rules are classified into four ranks, which are Most (M), Higher (H), Common(C), and least (L), and rules are applied in this order of ranks; Table 4 shows the dictionary ranks.

Table 4. Classified Rank of Rule

Name of Rank	Correct Prediction Rate (%)
Most (M)	CR≥90
High (H)	70≤CR<90
Common (C)	30≤CR<70
Least (L)	CR<30

The rule rankings are updated simultaneously with CF, EF, CR, and ER. The credibility evaluation function (CEF) of our proposed method is defined as follows:

$$\mathrm{CEF} = CR - \alpha ER + \beta P(A_{n+1} | A_n) \quad (5)$$

,where α, β are coefficients. $P(A_{n+1} | A_n)$ is the analyzed result that expresses the conditional contiguity action probability based on bigram. The rule credibility is high when $P(A_{n+1} | A_n)$ is high, CR is high and ER is low. The CR and ER are defined as:

$$CR = \frac{CF}{CF + EF} \quad (6)$$

$$ER = \frac{EF}{CF + EF} \quad (7)$$

The learning process consists of the following steps: acquisition of initial rules, rule's abstraction process, acquisition of action scripts, analysis of $P(A_{n+1} \mid A_n)$ with bigram, and rule's information updating. This implementation provides the possibility that the system is able to adapt dynamically to each user. When performing rule's abstraction process, in order to decide the criteria of common parts for natural element's states. We gave some range for each natural element. Time is 10 minutes, brightness is 50 Lux, and temperature is 2.

The rule's adaptive ability improves according to the number of iterations of the abstraction performed. If the abstracted rule is repeated excessively, the prediction accuracy will fall. For this reason, our system needs the suitable degree of abstraction. Since artificial elements and natural elements have unique aspects, we defined two degrees of abstraction.

$$Dan = \frac{X}{N} \times 100\% \qquad (8)$$

$$Daa = \frac{Y}{M} \times 100\% \qquad (9)$$

, where, *Dan* is the degree of abstraction for the natural elements in the rules, *X* is the number of variables in the natural elements portion, *N* is the number of natural elements, *Daa* is the degree of abstraction for the artificial elements in the rules, *Y* is the number of variables in the artificial elements portion. *M* is the number of artificial elements.

6. Evaluation Experiment

As mentioned above, the system based on the procedure shown in Figure 2 was developed for experimentation to investigate the validity of our proposed method.

In most machine learning experiments, cross-validation is the standard method of evaluating the performance of an algorithm with a single dataset of independent examples. If cross-validation is inappropriate, partitioning the data into separate training and test sets is usual. However since our aim is to propose an adaptive approach, we will evaluate online performance. It means our algorithm is tested on the current action using preceding actions for training. Therefore, we do not perform cross-validation evaluation. We divided data into training and testing data. After the learning process, including proofreading and feedback process test data becomes a new entry in training data, which means every new entry enlarges usable training data in our online system.

The data was collected based on the person's actual life, from five graduate students of Graduate School of Engineering, Hokkaido University. We generated about 16,000 data that corresponded to a period of 91 days.

Since our aim is to develop a dynamic adaptive prediction system for each user, in order to keep the starting state constant for each user, the data history and the rule dictionary always started from empty initial state.

In order to optimize the coefficients of the credibility evaluation function, we did some preliminary experiments. We applied the Greedy Method [10] to all collected data to obtain the optimal coefficients for the credibility evaluation functions. In the formula (5) the parameters α, β were found to be 10 and 20 respectively. For the abstracted rules, the value of *Dan* is 0.6 and *Daa* is 0.5 by preliminary experiments.

7. Results and Discussion

In this section we describe our experiments. In Figure 3 we plot the performance of our proposed method. We just use the top prediction result to evaluate our experiment's accuracy. It is shown that our method had good prediction performance.

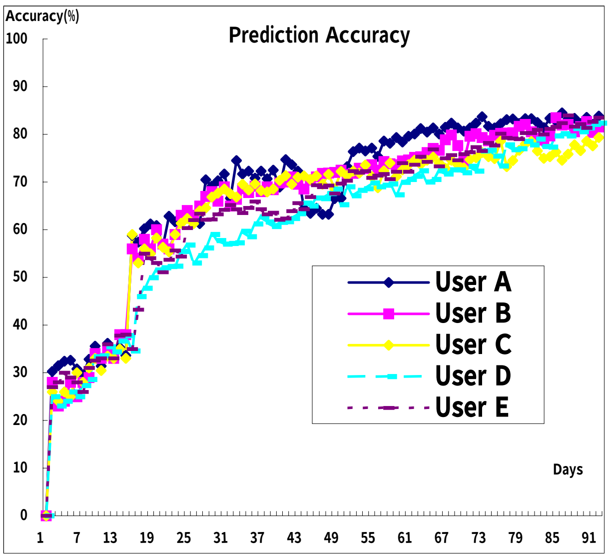

Figure 3. Transition of Prediction Accuracy

We found that the highest prediction accuracies for users are about 83.8% (A), 81.55% (B), 79.32% (C), 82.33% (D), and 83.47% (E). It means that the system based on our proposed method is able to acquire many immanent causality rules automatically from data pairs through IL. Being unified with N-gram, the information of user-performed actions can be enough for prediction.

This level of predictive accuracy obviates the need of preparing data history and rule dictionary in advance. The acquired rules express the user's habits and preferences. Therefore, the system can adapt itself dynamically to each user. The user needs to proofread the errors in prediction results, hence the prediction ability improves. As a result of the system's dynamic adaptation, the number of errors decreases.

8. Conclusions and Future Work

In this paper we have described our approach to predicting a user actions in an intelligent system such as a learning room. Our method differs from conventional statistical and analytical approaches. It is showed that our system performance has good accuracy for each user.

Furthermore, we also observed that our system could adapt dynamically to any user, since the data history and the rule dictionary were always initially empty. We did not need to prepare data history and rule dictionary beforehand.

We defined our proposed method as a Point-Pass-based Prediction System with the unification of IL and N-gram.

As future works, we will do some comparison experiments with the inductive reference (C4.5) and other method, and we hope to certify the validity of dynamic adaptation of our system to multiple users.

Reference

1. S.M. Weiss & N. Indurkhya: Rule-based Machine Learning Methods for Functional Prediction. Journal of Artificial Intelligence Research, Vol. 3, (1995), pp.383-403.
2. Davison B. Hirsh H. & Swami: A Probabilistic online Action Prediction, Proceedings of the AAAI Spring Symposium on Intelligent Environments, Technical Report SS-98-02, Stanford University, California, AAAI Press 1998, pp.148-154.
3. Z. Su, Q. Yang. & H. Zhang: Whatnext: A prediction system for web requests using n-gram sequence models, Proceedings of the First International Conference on Web Information System and Engineering Conference, Hong Kong, June 2000, pp.200-207.
4. S. Sato: A method for combining fragments of example-based translation. Artificial Intelligence, volume 75, May 1995, pp. 31-49.
5. K. Araki and K. Tochinai: Effectiveness of Natural Language Processing Method Using Inductive Learning Processing of the IASTED International Conference ARTIFICIAL INTELLIGENCE AND SOFT COMPUTING, May, 2001, Cancun, Mexico, pp.295-300.
6. H. Echizen-ya, K. Araki, Y, Momouchi & K. Tochinai: A Study of Performance Evaluation for GA-ILMT Using Travel English, Proceedings of the 13th Pacific Asia Conference on Language, Information and Computation, Taipei, Taiwan, February 1999, pp.285-292.
7. M. Matsuhara, K. Araki & K. Tochinai: Effectiveness for Machine Translation Method Using Inductive Learning on Number Representation, B. Mckay, J. Slaney (Eds.), AI2002: Advances in Artificial Intelligence, Lecture Note in Artificial Intelligence 2557, Springer-Verlag, December 2002, pp.648-659.
8. M. Matsuhara, K. Araki, Y. Momouchi. & K. Tochinai, Evaluation of Number-Kanji Translation Method Using Inductive Learning on E-mail, Proceedings of the IASTED International Conference Artificial Intelligence and Soft Computing, Banff, Canada, July 2000, pp. 487-493.
9. K. Araki, H.Echizen-ya. & K.Tochinai: Performance Evaluation in Travel English for GA-ILMT, Proceedings of the IASTED International Conference Artificial Intelligence and Soft Computing, Banff, Canada, July 1997, pp. 117-120.
10. T.Asano, &, H. Imai: Computation and Algorithms, Ohm Company 2000(in Japanese).