

Design of Machine Sorting Based College Course Selection System

Qiliang Sun^{1, a} and Qixia Zhang^{1, b *}

¹School of Mathematical Sciences, University of Jinan, Jinan 250022, China

^asuntsiang@163.com, ^bzhangqixia110@163.com

*The corresponding author

Keywords: Management information system; Linear regression; Least squares estimation; Java web; Sorting formula.

Abstract. This article discusses a machine sorting problem in Management Information System. We take course selection system as an example. Teachers browse the courses and apply for the courses on the web. According to machine sorting results, the system determines a list of teachers to teach the course. Among the design of course selection system, sorting is the most important content. Currently, the sorting process only takes into account four factors: teachers' professional titles, evaluation score, whether teachers have taught the course or not, needs of teachers. We choose linear regression model from historical data as our sorting formula. The sorting formula is applied for the design of course selection system. Machine sorting makes course arrangement more optimal, objective, fair, efficient, and economical. This WEB-based course selection system possesses a value of application and can be extended to the design and implementation of similar systems.

Introduction

There exists a management style in college which can be summarized as the following three aspects: teachers choose courses, teachers compete for courses, and students choose teachers. Specifically, every teacher arbitrarily selects courses according to their own characteristics firstly. Second, if there are multiple teachers select a course simultaneously, teachers need to be sorted [1]. Third, teachers teach courses for trial. Students are free to choose courses from different teachers, and then select one teacher as lecturer [2].

Following these rules and regulations, colleges and universities effectively prevent a teacher teaching in one or more courses repeatedly [3]. This ensures that teachers must constantly improve their teaching and researching level. Teachers can also update their courses according to their new fields of research [4].

This paper proposes a sorting method to design course selection system. According to the professional training programs, administrators enter all courses information includes member information, course information and evaluation score into the system [5]. Machine will choose a list of teachers as lecturers. The sorting formula is discussed in next section.

Sorting Formula

According to teachers' evaluation score and other information, we collect 77 samples and obtain the following teachers' information table.

Table 1 Teachers' information

Name	Whether the teacher has taught this course or not	Evaluation score	Professional titles	Course score
Zhang*	1	98.88648	4	77
Ma**	1	99.01011	3	76
Liu**	1	98.72676	3	75
Song**	1	98.54724	3	74
Miao**	1	98.00076	3	73
Sun**	1	97.05828	3	72
.....
Wu**	1	97.80001	2	3
Zhang*	0	97.92552	2	2
Li**	0	95.12448	2	1

We study the influence factors of the course score. Taking into account the availability of data and the convenience of the research, we believe that the main influence factors of the "course score" are "teachers' professional titles", "evaluation score", "whether teachers have taught the course or not" [6]. In Fact, The higher the teachers' professional titles, the higher the score is. The higher the evaluation score, the higher the course score is. So, we think that the explanatory variables should be "teachers' professional titles", "evaluation score", and "whether teachers have taught the course or not".

We choose the following linear regression model as sorting formula in our course selection system:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \varepsilon \quad (1)$$

Where y is the course score, x_1 is the evaluation score, x_2 is teachers' professional titles title, x_3 is "whether the teacher has taught the course or not", and ε is random error.

Using Eviews software to do the least squares regression, we get the following Eviews running results table:

Table 2 Eviews running results

Dependent Variable: SCORE

Method: Least Squares

Date: 07/12/15 Time: 14:08

Sample: 1 77

Variable	Coefficient	Std. Error	t-Statistic	Prob.
C	-521.7586	62.52919	-8.344242	0.0000
LEVAL	16.36305	0.931038	17.57506	0.0000
STARTED	44.01442	1.379321	31.91020	0.0000
EVA	5.160848	0.640211	8.061163	0.0001
R-squared	0.944849	Mean dependent var		39.00000
Adjusted R-squared	0.942582	S.D. dependent var		22.37186
S.E. of regression	5.360736	Akaike info criterion		6.246630
Sum squared resid	2097.837	Schwarz criterion		6.368386
Log likelihood	-236.4952	Hannan-Quinn criter.		6.295331
F-statistic	416.8789	Durbin-Watson stat		0.314548
Prob (F-statistic)	0.000000			

Then, our sorting formula is:

$$y = -521.7586 + 16.36305x_1 + 44.01442x_2 + 5.160848x_3 + \varepsilon \quad (2)$$

The above Sorting formula Eq. (2) shows that under the assumption that all other variables constant premise, 1 level increasing of teachers' professional titles, teacher has ever taught the course, and 1 point increasing of evaluation score will lead to 44.01442, 5.160848, and 16.36305 increasing of course score, respectively.

Moreover, statistical test in Table 2 shows that the model is very good to fit the sample.

System Design

System is based on Java Web SSH framework. With the increasing of data and functions, SSH framework speeds up the development process [7, 8].

The system focuses on three users: System administrators, teaching management workers, and regular teachers. Regular teachers can use the system to browse course information, apply for one or more courses, and view the results of course selection. Teaching management workers can perform the sorting by sorting formula. Basic functions of the system are as follows:

(1) Course information management: It includes adding, deleting, viewing, editing, and publishing new courses.

(2) Teacher information management: System administrators can add, delete, and manage regular members' accounts.

(3) Evaluation score management: It includes adding, deleting, viewing, and editing teachers' evaluation scores. Data are obtained from a teaching evaluation subsystem.

(4) Course management: Teachers can log in, browse the courses, and select one or more courses. Before the deadline, teachers can modify or cancel their selections at any time.

(5) Educational notification management: Teaching management workers can make announcements through the system. This is a general article management system that supports online modification [9].

In Microsoft SQL Server 2008R2 database, course and course selection tables are defined in the following:

Table 3 Course table definition

Name	Data Types	PK	FK	Comment
ID	int	√		primary key ID
CategoryID	int		√	course categories ID
cCode	char(8)			course code
cName	varchar(24)			course name
Campus	char(6)			school campus
Hours	int			hours required
Period	varchar(5)			period
Demands	int			teachers' requirement
Picture	varchar(100)			pictures of textbook
cDesc	text			description of courses
HtmlPath	varchar(100)			web address of courses
Textbook	varchar(50)			name of textbooks
Authors	varchar(50)			authors of textbooks
Publisher	varchar(30)			publisher of textbooks
Status	int			course publish true and false

Table 4 Course selection table definition

Name	Data Types	PK	FK	Comment
ID	int	√		primary key ID
SheetID	int		√	ID of courses
MemberID	int		√	ID of teachers
OrderID	int		√	ID of orders
Started	int			whether teachers have taught the course or not
Score	float			value need by machines in order to sort

Among them, the value of score in Table 4 is calculated by the sorting formula Eq.2. Class “RecordAction.java” implements sorting. Part of the code is in the following: “record.setScore(-521+44*Float.parseFloat(record.getStarted().toString())+16*Float.parseFloat(member.getMemberlevel().getId().toString())+5*record.getEvaluation().floatValue());”.

We choose Java language to development this project. Sequence diagram shows the interaction among objects [10]. The modify personal message sequence diagram figure is in the following:

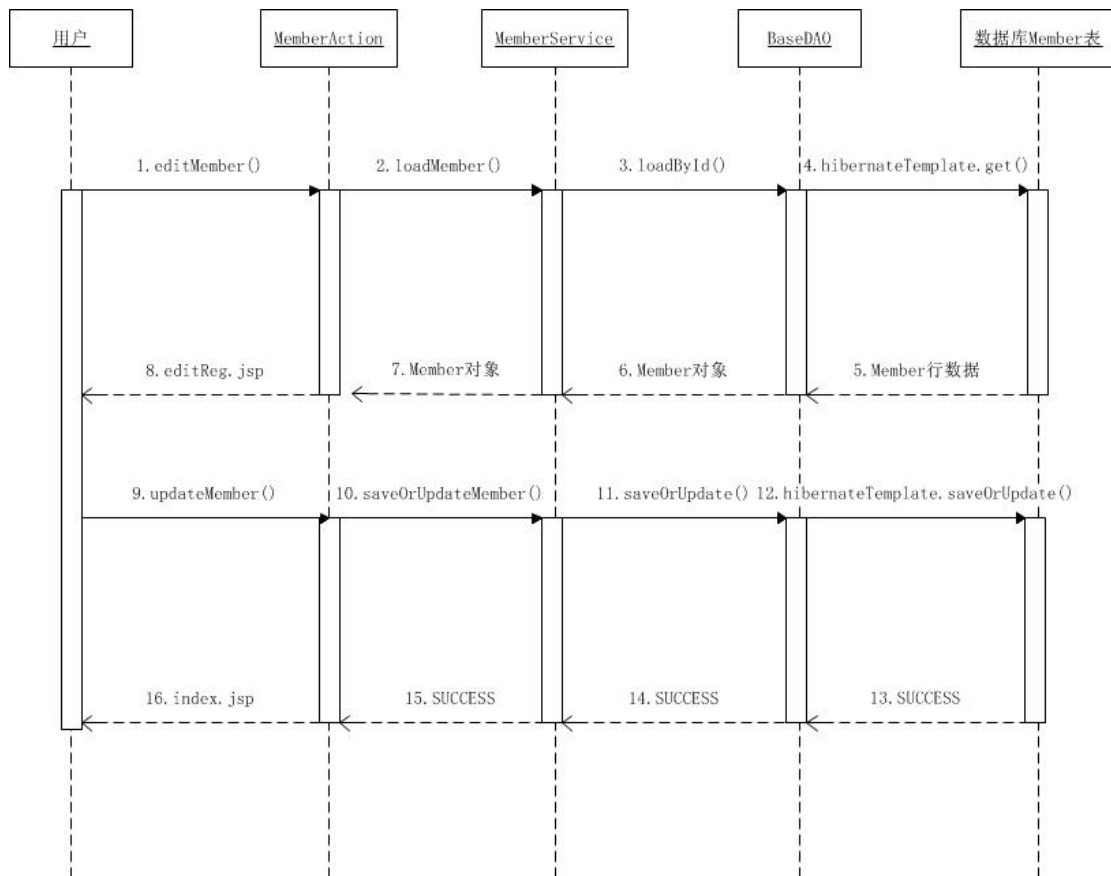


Figure 1. Modify personal message sequence diagram

Conclusions

The main work of this paper is that we introduce a sorting formula in design of college course selection system. Currently, the sorting process only takes into account four factors: teachers' professional titles, evaluation score, whether teachers have taught the course or not, needs of teachers. The system will automatically select the lists of lectures. A limitation of this design is that factors

considered in sorting formula are only four. In the future, other factors can be added to the sorting formula. For example, ages of teachers, educational backgrounds of teachers, and so on.

Acknowledgements

The authors acknowledge the support from the the Natural Science Foundation of Shandong Province, China (ZR2015AL012).

References

- [1] J. Alonso, R. Matias, E. Vicente, A. Maria and K.S. Trivedi: Performance Evaluation, Vol. 70 (2011) No.3, p.231.
- [2] G.R. Ning, K.S. Trivedi, H. Hu and K.Y. Cai: *IEEE 3rd International Workshop on Software Aging and Rejuvenation* (Hiroshima, Japan, November, 2011).p.32.
- [3] R. Matos, J. Araujo, V. Alves and P. Maciel: *IEEE International Conference on System* (Seoul, Korea, October, 2012).p.1103.
- [4] J. Araujo, R. Matos, P. Maciel and R. Matias: *IEEE International Conference on System, Man, and Cybernetics* (Anchorage, United states, October, 2011).p.1411.
- [5] M. Mengoni, S. Graziosi, M. Mandolini and M. Peruzzini: *International Journal of Interact Design and Manufacturing*, Vol. 5 (2011), p.1.
- [6] K.D. Bouzakis, G. Andreadis, A. Vakali and M. Sarigiannidou: *Advances in Engineering Software*, Vol. 40 (2009), p. 956.
- [7] D. Cotroneo, S. Orlando and S. Russo: *26th IEEE International Symposium on Reliable Distributed Systems* (Beijing, China, October 2007).p.127.
- [8] J. Alonso, J. Torres, J.L. Berral and R. Gavalda: *International Symposium on Parallel & Distributed Processing, Workshops and Phd Forum* (Atlanta, United states, April, 2010).p.1.
- [9] H.J. Lee, J.W. Lee and J.O. Lee: *Advances in Engineering Software*, Vol. 40 (2009), p.176.
- [10] H.B. Wang, J.Z. Huang, Y.Z. Qu and J.Y. Xie: *Journal of Web Semantics*, Vol. 1 (2004), p.309.