Book Recommendation Using Machine Learning Methods Based on Library Loan Records and Bibliographic Information

Keita Tsuji [1)], Fuyuki Yoshikane [2)], Sho Sato [3)] and Hiroshi Itsumura [4)]

Faculty of Library, Information and Media Science,
University of Tsukuba
1-2 Kasuga, Tsukuba-city, Ibaraki 305-8550, Japan
{keita[1)], fuyuki[2)], hits[4)]}@slis.tsukuba.ac.jp

Faculty of Social Studies, Doshisha University
Karasuma Higashi-iru, Imadegawa-dori,
Kamigyo-ku, Kyoto 602-8580, Japan
min2fly@gmail.com[3)]

***Abstract*—We propose a method to recommend books through machine learning modules based on several features, including library loan records. We evaluated the most effective method among ones using (a) a Support Vector Machine (SVM), (b) Random Forest and (c) Adaboost, as well as the most effective combination of relevant features among (1) library loan records, (2) book titles, (3) Nippon Decimal Classification categories, (4) publication year and (5) frequencies at which books were borrowed. We performed an experiment involving 40 subjects who are students at T University. The books that our methods recommended and the loan records that we used were obtained from the T University Library. The results show that books recommended by the SVM based on features (1), (2), (3) and (5) were rated most favorably by the subjects. Our method outperforms preceding ones, such as the method proposed by Tsuji et al. (2013), and is comparable in performance to the recommendation by the website Amazon.co.jp.**

Keywords— Book Recommendation; Recommender System; Library Loan Records; Support Vector Machine (SVM); Random Forest; Adaboost

I. INTRODUCTION

In this paper, we propose a method to recommend books through machine learning modules based on several features, including library loan records. While several methods for automated book recommendation have been proposed in the literature, the effectiveness of these methods has not been researched very thoroughly. In this study, we examined the effectiveness of book recommendation methods that use Support Vector Machines (SVMs), Random Forest, and Adaboost modules. Specifically, we evaluated the effectiveness of each of the following features to recommend books: (1) library loan records, (2) book titles, (3) Nippon Decimal Classification (NDC) categories, (4) publication year and (5) the number of times a book has been borrowed.

When browsing a book on Amazon.co.jp (henceforth "Amazon"), the website recommends books to the user by listing other titles purchased by users who also bought the text that the user is browsing as "Customers who bought this item (i.e., book) also bought ..." An OPAC (online public access catalog) that recommends books according to Amazon (under the Amazon's permission) might be a cost-effective recommendation system compared to the one that recommends books based on machine learning and library loan records. Based on this background, we investigated (a) which machine learning method among the ones listed above is the most effective, (b) which features are useful in making appropriate recommendations and (c) whether our method can outperform Amazon's recommendation system.

We assume that the method is incorporated into the OPAC of university libraries. It recommends books to students when they show interest in specific texts (for instance, by clicking a book title).

II. RELATED STUDIES

Little research has been conducted on book recommendation systems for libraries (Whitney & Schiff (2006), Chen & Chen (2007), Luo et al. (2009), Harada (2009), Harada & Masuda (2010), Shirgaonkar et al. (2010), Tsuji et al. (2012) and Tsuji et al. (2013)). However, studies that have been carried out have shown that the association rule is more effective than collaborative filtering (Tsuji et al. (2012)), and that an SVM based on (1) the association rule, (2) book titles and (3) NDC categories is more effective than one based only on the association rule (Tsuji et al. (2013)).

III. DATA

A. Library Loan Records and Recommended Books

For this study, we used 2,293,642 loan records in the T University library. The checkout dates for these range from January 2, 2006 to March 31, 2012. The same data was used by Tsuji et al. (2013). For books to recommend, we used 643,676 books in the T University Library (again, this is the same data used by Tsuji et al. (2013)).

B. Subjects and Training Data

Forty students majoring in library and information sciences at T University participated as subjects in our experiment, consisting of 15 graduate students, 11 fourth-year

978-1-4799-4173-5/14 $31.00 © 2014 IEEE
DOI 10.1109/IIAI-AAI.2014.26

undergraduate students, and 14 second-year undergraduate students.

C. A Book the Subject would Like to Borrow

Subjects were asked to provide the title (and other bibliographic information, if necessary) of one book that they would like to borrow from the T University Library at present for research or study purposes (henceforth, "a book that currently interests the subject"). This information was used to recommend books based on association rules, similarities between the titles, matches between the NDC categories, and the Amazon recommendation system.

D. Training Data

We used 358 pairs of books as training data, the same as that used in Tsuji et al. (2013).

IV. Recommendation Method

A. Recommendation by Machine Learning Methods

We adopted the SVM, Random Forest, and Adaboost machine learning methods, as they are the most widely used. We used the following five sources of information as features for machine learning:

(a) Confidence and support of association rules between candidate books and the "book that currently interests the subject," extracted from the library loan records

(b) Similarities in titles between candidate books and the "book that currently interests the subject"

(c) Matches/mismatches in the NDC category of candidate books and the "book that currently interests the subject" (if all of their classes, divisions, and sections match, then 1; else 0)

(d) Publication years of the candidate books

(e) Frequency at which the books have been borrowed

For details of factors (a), (b) and (c), see Tsuji et al. (2013). As for (d), we assigned to each book a natural logarithm of "2014 minus its publication year." For instance, we assigned ln(6) to a book that was published in 2008. We tried to use publication years directly, however, the above-mentioned logarithm seemed to be more effective in our pilot examination. As for (e), we first counted how many times each book was borrowed by the students and faculties majoring in library and information science during the period of our library loan records (i.e., from January 2, 2006 to March 31, 2012). Then, we assigned to each book $\ln(N / Y + 1)$ where Y is defined as follows: (a) "2013 minus the publication year of the book" if the book was published in or after 2006 or (b) seven (i.e. 2013 minus 2006) if the book was published before 2006. For instance, we assigned $\ln(20 / 5 + 1)=\ln(5)$ to a book that was published in 2008 and was borrowed 20 times during the previously mentioned period. This is a natural logarithm of a rough approximation of the frequency at which the book was borrowed per year.

For an SVM, we used LIBSVM ver. 3.12, and adopted the L1 soft margin SVM "C-Support Vector Classification" and the Radial Basis Function (RBF) kernel. We used the easy.py script to obtain the optimal parameters C (the margin parameter that determines the generalization ability) and γ (the parameter that determines the influence of a single training example). We also used –b option to display the probability that the book interested the subject (more precisely, probability that the book belonged to the positive examples in the training data that had interested the past subjects).

For Random Forest, we used the package "randomForest" of R, a free software environment for statistical computing and graphics. First, we used the "tuneRF" function to tune the "mtry" parameter (the number of variables randomly sampled as candidates at each split) based on the training data. Then, we used the "randomForest" function to learn and the "predict" function to classify testing data and display their probabilities.

For Adaboost, we used the package "ada" in R. We used the "ada" function for the module to learn and the "predict" function to classify testing data and display their probabilities. We chose discrete boosting and performed 50 boosting iterations. Both these options are part of the default settings of the "ada" function.

We selected six books whose probabilities were the highest for the SVM, Random Forest, and Adaboost, based on combinations of features (a)(b)(c), (a)(b)(c)(d), (a)(b)(c)(e) and (a)(b)(c)(d)(e) described above, and recommended these books to subjects. Therefore, we recommended to each subject 6 (books) × 3 (machine learning methods) × 4 (combinations of features) = 72 books. Certainly, there are many duplicate results among the recommendation methods, because of which the actual number of recommended books is smaller than 72, and different for each subject.

Apart from machine learning, we recommended six books whose NDCs were the same as that of "book that currently interests the subject" and the title similarities to them were the highest (henceforth, we represent this method as "NdcTitle").

B. Amazon

We located "a book that currently interests the subject" from Amazon, chose books from Amazon's recommendation list sequentially starting from the left, and checked the OPAC to see if the T University Library had the recommended books. By doing so, we recommended to subjects six books that were recommended by Amazon and available at the T University Library (for brevity, we will call these books "those recommended by Amazon").

V. Evaluation Method

The bibliographic information for the recommended books was shown to the subjects. The subjects were then asked to describe their level of interest in each book using the following five-point scale also used by Tsuji et al. (2013): "2: Very Interested," "1: Interested," "0: Not Interested," "x: Have No Idea," and "A: Have Already Bought or Read."

VI. RESULTS

A. Overall Results

Table I shows the evaluation results for recommendation based on: (1) SVM, (2) Random Forest, (3) Adaboost, (4) NdcTitle and (5) Amazon. "SVM-APL" represents the results by SVM based on (a) confidence and support of association rules, (b) title similarities, (c) NDC matches, (d) publication year, and (e) the frequencies with which the books were borrowed. "SVM-AxL," "SVM-APx" and "SVM-Axx" represent the results for SVM based on the combinations (a)(b)(c)(e), (a)(b)(c)(d) and (a)(b)(c) of features, respectively. "RnF" and "Ada" represent the results using Random Forest and Adaboost, respectively, and the APL, AxL, APx, and Axx for each of these represent analogous features to those in the case of SVM.

Table I shows that 240 books were recommended by "SVM-APL." Of these, 67 were evaluated as "2: Very Interested" by the subject, accounting for 27.9% (= 67 / 240 × 100) of the recommended books. On the other hand, 150 books were recommended by Amazon, of which 64 were evaluated as "1: Interested," hence accounting for 42.7% (= 64 / 150 × 100) of all recommendations.

If "A: Already Bought or Read," "2: Very Interested," and "1: Interested" are considered to be "positive evaluations" – as they were in the report of Tsuji et al. (2013) – the method (excluding Amazon) with the highest percentage of positive evaluations (henceforth "PPE") is "SVM-AxL" with 80.0% of its recommendations receiving a positive evaluation (= (16 + 65 + 111) / 240). This is higher than 71.3% for "SVM-Axx" (= (7 + 51 + 113) / 240), and the difference is statistically significant at 0.05. Tsuji et al. (2013) had concluded that "SVM-Axx" is the most effective book recommendation method. In contrast, our method incorporates "frequencies at which the books were borrowed" into the features taken into account by the SVM, and is thus more effective than the method proposed by Tsuji et al. (2013).

Recommendations made by Amazon have an 82.7% PPE (= (20 + 40 + 64) / 150). While this is higher than the PPE for "SVM-AxL," the difference is not significant at 0.05. Note that "SVM-AxL" recommended 240 books while Amazon recommended only 150. If PPEs for two methods are similar, the method that can recommend more books is preferable. In this sense, "SVM-AxL" seems to be more effective than Amazon. We will discuss this point again in the next section.

Finally, the PPE of "NdcTitle" was 73.3% (= (6 + 54 + 116) / 240). As previously mentioned, "NdcTitle" does not use machine learning and makes recommendations based only on NDC matches and title similarities. Therefore, we see that machine learning is more effective than a simpler method.

B. Results by Probabilities

SVM, Random Forest, and Adaboost assign a probability (to each book) that a book interests the subject. If we only recommend books that are likely to interest subjects, the result will change. Based on this idea, we divided the recommended books into three groups: books whose probabilities are (1) greater than or equal to 0.9, (2) no less than 0.8 and less than 0.9, and (3) less than 0.8. The results of (1) are shown in Table II.

We see in Table II that the percentage of positive evaluation (PPE) for 146 books recommended by "SVM-AxL" is 84.2%. As previously mentioned, the PPE for 150 books recommended by Amazon is 82.7% (in Table I). Therefore, "SVM-AxL" is comparable to Amazon when we recommend books whose probabilities are no less than 0.9.

The PPE for "SVM-APL" is slightly higher than that for "SVM-AxL." However, the number of books recommended is only 83, which is much smaller than the number of books recommended by "SVM-AxL." With regard to other methods, such as "RnF-APL" and "RnF-AxL," either the number of books recommended is small or the percentage of positive evaluation is low for each of them.

C. Results by Grade

The results were compiled by dividing the subjects into second-year, fourth-year undergraduate students, and graduate students, as shown in Table III. As the table shows, for second-year, fourth-year undergraduate students and graduate students, the PPE was the highest for "SVM-AxL" at 91.7%, 72.7% and 74.4% respectively. Thus, "SVM-AxL" seems to be the most effective method regardless of the students' grades.

VII. CONCLUSIONS

We proposed a method to recommend books through machine learning modules based on several features, including library loan records. We showed that the method that uses SVM based on (1) association rules derived from library loan records, (2) similarities of book titles, (3) matches/mismatches between NDC categories and (4) the number of times a book has been borrowed performs better than (a) methods, such as Random Forest and Adaboost, based on the same features, and (b) SVMs based on other combinations of features. Furthermore, our method outperformed the ones proposed in previous studies, such as Tsuji et al. (2013), and is comparable to Amazon's method of book recommendation.

In the future, we aim to examine (1) the effectiveness of incorporating into recommendation methods information specific to university students, such as courses that they are taking, and (2) the effect of the size of the learning data on the performance of different methods.

REFERENCES

Breiman, Leo (2001) "Random Forests," Machine Learning, 45(1), 5-32.

Chen, C., & Chen, A. (2007). Using data mining technology to provide a recommendation service in the digital library. The Electronic Library, 25(6), 711-724.

Friedman, Jerome, Hastie, Trevor and Tibshirani, Robert (2000) "Additive Logistic Regression: A Statistical View of Boosting," The Annals of Statistics, 28(2), 337-407.

Harada, T. (2009). The book recommendation system using library loan records. Digital Libraries, 36, 22-31 (text in Japanese).

Harada, T., & Masuda, K. (2010). A trial approach of weighting for library loan records for developing a book recommendation system. Digital Libraries, 38, 54-66 (text in Japanese).

Luo, Y., Le, J., & Chen, H. (2009). A privacy-preserving book recommendation model based on multi-agent. Proceedings of the 2009 Second International Workshop on Computer Science and Engineering, 323-327.

LIBSVM and easy.py <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>. [Accessed: May 7, 2014]

MeCab <https://code.google.com/p/mecab/>. [Accessed: May 7, 2013]

R <http://www.r-project.org/>. [Accessed: May 7, 2014]

Shirgaonkar, S., Rajkumar, T., & Singh, V., (2010). Application of improved a priori in university library. International Conference and Workshop on Emerging Trends in Technology (ICWET 2010), 535-540.

Tsuji et al. (2012) Use of Library Loan Records for Book Recommendation. Proceedings of the 3rd IIAI International Conference on e-Services and Knowledge Management (IIAI ESKM 2012), 30-35.

Tsuji, et al. (2013). Book Recommendation based on Library Loan Records and Bibliographic Information. Proceedings of the 3rd International Conference on Integrated Information (IC-ININFO 2013). 8p. (No Pagination).

Whitney, C., & Schiff, L. (2006). Melvil Recommender Project: Developing library recommendation services. D-Lib Magazine, 12(2). <http://www.dlib.org/dlib/december06/whitney/12whitney.html>. [Accessed: May 7, 2014]

TABLE I. OVERALL RESULTS

	PPE	A: Already Bought or Read	2: Very Interested	1: Interested	0: Not Insterested	x: Have No Idea	Total
SVM-APL	77.5	16 (6.7)	67 (27.9)	103 (42.9)	51 (21.3)	3 (1.3)	240
SVM-AxL	80.0	16 (6.7)	65 (27.1)	111 (46.3)	44 (18.3)	4 (1.7)	240
SVM-APx	70.8	3 (1.3)	57 (23.8)	110 (45.8)	68 (28.3)	2 (0.8)	240
SVM-Axx	71.3	7 (2.9)	51 (21.3)	113 (47.1)	60 (25.0)	9 (3.8)	240
RnF-APL	63.3	8 (3.3)	44 (18.3)	100 (41.7)	84 (35.0)	4 (1.7)	240
RnF-AxL	57.1	6 (2.5)	48 (20.0)	83 (34.6)	100 (41.7)	3 (1.3)	240
RnF-APx	55.4	1 (0.4)	50 (20.8)	82 (34.2)	103 (42.9)	4 (1.7)	240
RnF-Axx	57.9	4 (1.7)	43 (17.9)	92 (38.3)	92 (38.3)	9 (3.8)	240
Ada-APL	63.8	5 (2.1)	50 (20.8)	98 (40.8)	80 (33.3)	7 (2.9)	240
Ada-AxL	65.8	10 (4.2)	48 (20.0)	100 (41.7)	75 (31.3)	7 (2.9)	240
Ada-APx	63.8	3 (1.3)	45 (18.8)	105 (43.8)	84 (35.0)	3 (1.3)	240
Ada-Axx	61.7	6 (2.5)	48 (20.0)	94 (39.2)	84 (35.0)	8 (3.3)	240
NdcTitle	73.3	6 (2.5)	54 (22.5)	116 (48.3)	52 (21.7)	12 (5.0)	240
Amazon	82.7	20 (13.3)	40 (26.7)	64 (42.7)	24 (16.0)	2 (1.3)	150

TABLE II. RESULTS FOR BOOKS WHOSE PROBABILITIES ARE 0.9 OR MORE

		PPE	A: Already Bought or Read	2: Very Interested	1: Interested	0, x: Not Interested	Total
0.9～	SVM-APL	85.5	4 (4.8)	29 (34.9)	38 (45.8)	12 (14.5)	83
	SVM-AxL	84.2	7 (4.8)	44 (30.1)	72 (49.3)	23 (15.8)	146
	SVM-APx	—	0 (—)	0 (—)	0 (—)	0 (—)	0
	SVM-Axx	—	0 (—)	0 (—)	0 (—)	0 (—)	0
	RnF-APL	81.3	0 (0.0)	6 (18.8)	20 (62.5)	6 (18.8)	32
	RnF-AxL	65.0	3 (2.9)	27 (26.2)	37 (35.9)	36 (35.0)	103
	RnF-APx	51.9	0 (0.0)	11 (20.4)	17 (31.5)	26 (48.1)	54
	RnF-Axx	70.4	1 (3.7)	6 (22.2)	12 (44.4)	8 (29.6)	27
	Ada-APL	64.5	0 (0.0)	8 (25.8)	12 (38.7)	11 (35.5)	31
	Ada-AxL	—	0 (—)	0 (—)	0 (—)	0 (—)	0
	Ada-APx	—	0 (—)	0 (—)	0 (—)	0 (—)	0
	Ada-Axx	—	0 (—)	0 (—)	0 (—)	0 (—)	0

TABLE III. RESULTS BY GRADE

		PPE	A: Already Bought or Read	2: Very Interested	1: Interested	0, x: Not Interested	Total
Second-Year Undergraduates	SVM-APL	88.1	3 (3.6)	32 (38.1)	39 (46.4)	10 (11.9)	84
	SVM-AxL	91.7	2 (2.4)	32 (38.1)	43 (51.2)	7 (8.3)	84
	SVM-APx	79.8	0 (0.0)	28 (33.3)	39 (46.4)	17 (20.2)	84
	SVM-Axx	71.4	1 (1.2)	24 (28.6)	35 (41.7)	24 (28.6)	84
Fourth-Year Undergraduates	SVM-APL	71.2	1 (1.5)	19 (28.8)	27 (40.9)	19 (28.8)	66
	SVM-AxL	72.7	1 (1.5)	20 (30.3)	27 (40.9)	18 (27.3)	66
	SVM-APx	71.2	2 (3.0)	18 (27.3)	27 (40.9)	19 (28.8)	66
	SVM-Axx	66.7	2 (3.0)	14 (21.2)	28 (42.4)	22 (33.3)	66
Graduate Students	SVM-APL	72.2	12 (13.3)	16 (17.8)	37 (41.1)	25 (27.8)	90
	SVM-AxL	74.4	13 (14.4)	13 (14.4)	41 (45.6)	23 (25.6)	90
	SVM-APx	62.2	1 (1.1)	11 (12.2)	44 (48.9)	34 (37.8)	90
	SVM-Axx	74.4	4 (4.4)	13 (14.4)	50 (55.6)	23 (25.6)	90