

This item was submitted to Loughborough's Institutional Repository (<https://dspace.lboro.ac.uk/>) by the author and is made available under the following Creative Commons Licence conditions.



For the full text of this licence, please go to:
<http://creativecommons.org/licenses/by-nc-nd/2.5/>

Automating Library Stock Ordering from Reading Lists

J.P. Knight, G.P. Brewerton, J.L. Cooper

July 29, 2013

Abstract

Deciding which works should be held by a university library, and how many copies of each work are required, is a time consuming process. Library staff have to consider a number of variables in making each purchasing decision, and use data from a variety of library systems. At Loughborough University one such system is the online reading lists, which allows academic staff to indicate to students and library staff which books they feel are most useful/important. This paper discusses taking data from the reading list system, and other library data sources, and using it in an automated system to make book purchasing suggestions to ease this task for library staff.

1 INTRODUCTION

Collection development is one of the major tasks of any library. In higher education libraries works used to support learning and teaching are of great importance, and much effort is expended by library staff in ensuring that there are adequate numbers of suitable books available to the students. For every book that is purchased, library staff have to consider a number of variables and many of these are based on data gleaned from existing databases and services that the library runs.

At Loughborough University, learning and teaching has long been supported by the University Library's online reading list system. This provides students with a web based view of their reading lists, and can be kept up to date by the academics themselves, again using their web browsers. Library staff make use of the reading list system to check that essential and important books are in stock, and to help ensure that enough copies are available based on the number of students on each module.

Until now working out how many copies of individual titles should be purchased has been a tiresome, manual operation for library staff. They have had to systematically go through all the reading lists that they monitor, find how many students are on the module that the reading list applies to and then for each book on that reading list check to see what level of importance the academic has given it, how many copies are currently in stock (if any) and how many other modules the same book is on. Purchasing decisions then have to be made based on this data, coupled with the funds available in different departments' budgets (as some works are used by students on more than one module from

more than one department). Sometimes different library staff can be working on the same work at the same time as it appears in different modules, resulting in duplication of effort.

With this in mind a plan was made to extend the reading list system to help library staff in making acquisition decisions. The idea was to bring together the data that the staff were already using, most of which was being extracted manually from various online data sources, and then codify the purchasing decisions as much as possible. The result would be a set of suggestions that library staff could then use to make purchasing easier.

An initial “proof of concept” prototype system was produced to demonstrate that it could acquire the required data and then use it to make purchasing recommendations. The output of this prototype system was then used to make purchasing suggestions for a bulk purchase of teaching support materials, with the resulting suggestions submitted to both library staff and academic departments for checking.

2 SYSTEM DESIGN

The system described in this paper is an extension to the Loughborough Online Reading List System (LORLS) [1]. LORLS version 6 is a complete rewrite of the LORLS system taking into account the lessons learnt from over a decade of reading list use at Loughborough and elsewhere [2]. LORLS version 6 is designed to be a flexible system that can accommodate a variety of organisational structures and material types. As with the previous versions of LORLS it is web delivered, but unlike earlier versions it makes heavy use of “Asynchronous Javascript and XML” (AJAX) [3] interactions in order to provide a variety of rich user front ends to the backend database system. LORLS is available as open sourced software under the GNU General Public Licence version 2 [4] and has been used at a number of different institutions. It can be downloaded from <http://blog.lboro.ac.uk/lorls/downloads>. The LORLS database can immediately provide some of the information that we require for our automated purchasing recommendations: The system described in this paper is an extension to the Loughborough Online Reading List System (LORLS) [1]. LORLS version 6 is a complete rewrite of the LORLS system taking into account the lessons learnt from over a decade of reading list use at Loughborough and elsewhere [2]. LORLS version 6 is designed to be a flexible system that can accommodate a variety of organisational structures and material types. As with the previous versions of LORLS it is web delivered, but unlike earlier versions it makes heavy use of “Asynchronous Javascript and XML” (AJAX) [3] interactions in order to provide a variety of rich user front ends to the backend database system. LORLS is available as open sourced software under the GNU General Public Licence version 2 [4] and has been used at a number of different institutions. It can be downloaded from <http://blog.lboro.ac.uk/lorls/downloads>.

The LORLS database can immediately provide some of the information that we require for our automated purchasing recommendations:

- Module information (module code, module title, number of students registered on the module, etc), some of which is itself extracted from the University’s central administration system,

- Which reading list(s) are associated with each module,
- Details of individual works on each list, along with the level of recommendation for each item as suggested by the academic (for example “Essential”, “Important” or “Recommended”).

The Aleph Library Management System (LMS) database can also provide further useful information:

- Number of existing items (copies) of a work,
- The loan type for each item (long loan, week loan, short loan, reference, etc),
- The costs associated with previous item purchases for a work,
- The budgets available for purchasing new material for each department,
- Loan histories for items.

With these data sources available investigations could begin into combining the information they provided into the prototype “proof of concept” purchasing predictor. The basic flow chart for the system is shown in Figure 1.

When given a set of ISBNs (either explicitly or by looking at the ISBNs of works on reading lists associated with a particular module, or on modules in a given department, or all modules held in the system) the first task is to find out how many copies (or “items”) the library holds of each of the works, and what loan type in the LMS each item has. We need the latter as some loan types such as “reference” are not actually loanable. Next an assessment is required for the number of copies needed in an ideal world to satisfy demand. This assessment is made for each ISBN based on a “student ratio”, which is the number of students expected to share a single copy of a work. This ratio will be set for the whole institution, with the ability to override it for individual departments and possibly even courses.

There is also the need for student ratios to be different depending on the level of importance that an academic has attached to a particular work. For example if there are 50 students on a module with two works on the reading list, the library would want to have more copies of the work that is marked as “essential” compared to the one that is simply background reading. The appropriate ratio is multiplied by the number of students on each module that the work is on, to give a total number of copies of a work required for that module. This is done for every module that the work appears on, across all departments.

Some institutions may want to put in a maximum upper limit on the number of copies that will be purchased for any one work. This may be done to limit the expenditure on particular classes of text for example. At Loughborough this has traditionally been set at 35 copies for any one work.

Once the system has ascertained how copies are required for a work, it checks if there are already the required number of copies held by the library and discounts such works from further processing.

Of the works remaining, it then looks at the utilisation of the copies currently in circulation. This is necessary because there may be some works that are only borrowed occasionally by students throughout their module, whereas others

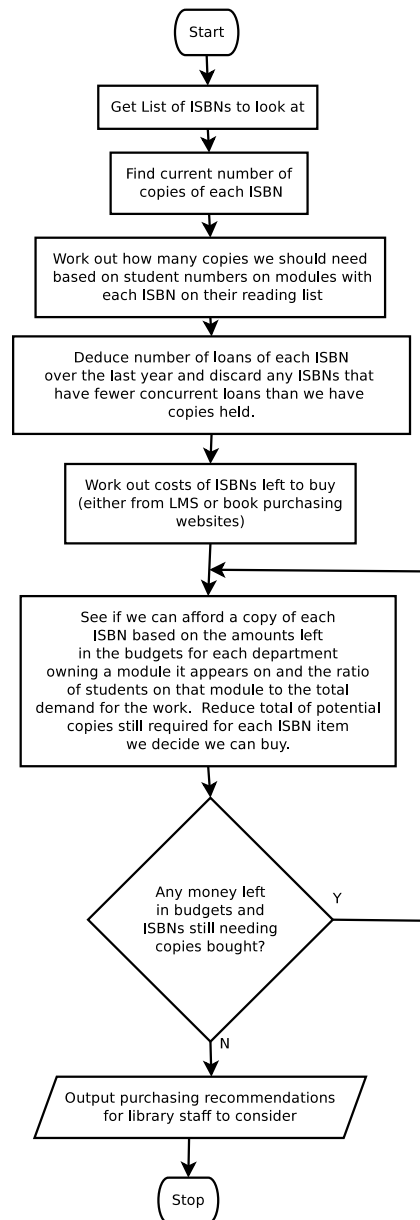


Figure 1: Flowchart showing the logic underlying the prototype system.

have large numbers of students all wanting to borrow copies to meet a project deadline at the same time.

The system aims to buy additional copies of works where a given fraction of the existing copies have been loaned out simultaneously in the last year. This fraction is another parameter (ranging from 0 to 1) that can be adjusted in the purchasing decision. If set to 1 it will only try to buy works that have had all available items loaned out at the same time. Again, works that do not meet this criteria are discounted from further processing.

The works that remain after the loan analysis are those that potentially need purchasing to satisfy student demand. However, the system still needs to work out if the library can afford to purchase them, and also determine how many copies of each work can be purchased (for those works that require more than one copy to be purchased to satisfy student demand).

To do this the program first needs to work out the cost of an individual copy of each work. This is done in two ways. If any copies have been bought in the past the system takes the maximum cost of any copy recorded in the Library management system. If this is a new work, the system tries to retrieve the average price from third party sources. These third party sources include isbndb.com, Google Books and Bookfinder.com.

The results of third party pricing lookups is cached by the system so that repeated runs don't hit these services too hard, especially as some have a cost per ISBN lookup (such as isbndb.com).

The system then sorts the remaining works based on the number of copies it has predicted are required, so that high demand works are considered first. This is important as, if budgets are limited, the system needs to purchase the most heavily over subscribed works first, before funds start to run out.

The program then checks if there are sufficient funds to purchase one copy of each. The funds are based on the balances left in the purchasing budgets allocated to each department in the LMS, plus their allowed "overspend". The proportion attributed to each budget is based on the number of modules run by each department that use this work, and the number of students on those modules. Departments with large numbers of modules using a work and/or modules with large numbers of students on them will take a greater share of the cost for each copy than a department that only includes the work in the reading list for one module taken by a handful of students. This also means that individual departmental purchasing budgets may be charged for fractions of a copy, and each copy of a work may well be paid for from more than one departmental budget.

For every work processed we either deduct one from the number of copies still needing to be bought or, if the available funds have run out, we remove that work from further processing. Once all the works have been investigated we loop round again for another pass, and continue doing this until we either have no works left that need copies bought or no longer have enough funds for further purchases left in the budget.

At this point the system outputs a list of ISBNs for works that should be purchased and the number of copies (or parts of copies) to be allocated to each departmental budget. This can then be used by library staff as a guide for which works should now be purchased via the LMS.

Table 1: Departmental purchasing suggestion cost summary

Department	Amount
English and Drama	£15,352.25
Politics, History and International Relations	£7,708.57
Social Sciences	£5,552.96
Geography	£3,544.08
Information Science	£2,611.61
Manufacturing and Mechanical Engineering	£2,404.43
Civil Engineering	£2,028.36
Business School	£1,571.66
Art	£1,423.22
Design School	£1,423.22
Electrical Engineering	£1,384.61
Chemical Engineering	£1,179.62
Materials Science	£730.56
Aeronautical and Automotive Engineering	£633.62
Chemistry	£538.68
Physics	£460.22
Mathematics	£407.74
Economics	£393.98
Sport, Exercise and Health Sciences	£347.18
Computer Science	£302.82
Total	£49,999.39

3 PROOF OF CONCEPT TESTING

The proof of concept prototype system was tested against Loughborough University Library’s live reading list system and LMS. During the 2012 summer vacation the university library was given a budget of £50,000 to improve the provision of books for teaching across the university. We decided to use this as an opportunity to test the purchase predictor algorithms.

Running the purchase predictor across all the works in all the active modules taught across the whole university took many hours on each run, although caching the results of pricing look ups and loan histories helped to improve performance. The results were presented as a web page and we prevented the web browser from timing out by utilising a dynamic progress bar that was updated constantly throughout the purchase prediction process.

The purchase predictor finally suggested buying 789 copies of 626 works costing £49,999.39. The break down of costs by department is shown in Table 1.

The departments at Loughborough University are split into ten Schools, and the detailed list of suggest purchases for each department was sent to the appropriate library liaisons. Of those that replied:

- Ten said the suggestions for their departments were broadly fine and they were happy to proceed with them.
- Four said it was mostly fine, but asked for a few minor changes to be considered.
- One department said that it reflected their research rather than their teaching as that was what their academics had put on their reading lists to date. However they had agreed a different set of “basic readers” for their undergraduate course which they suggested instead. It was pointed out that they should really update their online reading lists to reflect this change.

Some of the comments we received back from the library liaisons in the academic departments included:

- “I have reviewed the list of recommended purchases and it seems like a reasonable selection to me.”
- “Those suggestions sound good to me.”
- “I have reviewed the lists, and I think the titles are all perfectly acceptable.”
- “The list proposes to buy the older 2007 second edition [...] However, I was wondering if I could instead have 3 copies of the newer 2012 third edition which has recently been published?”
- “If the library has already done an analysis of borrowing patterns and established priority needs, then I think we as a department would be happy to go with your recommendations for now.”

These department responses were broadly in line with what we would expect to receive if the suggestions had been compiled by hand by the library staff. This demonstrated that the automated purchasing predictor algorithms were generating sensible suggestions. Indeed some of the minor changes requested were because new paper books and ebooks had been acquired in the period between the purchase predictor code running and the lists being distributed to the departments.

There were also some academic staff who were starting new modules or taking over existing modules from colleagues, and not all of these had completed their reading lists. In those cases they sometimes suggested additional purchases that they would find useful, and its was also a reminder to them that they needed to ensure that the reading list system was updated promptly if they didn’t want to miss out on having enough suitable books for their students from the Summer purchasing round

4 FURTHER DEVELOPMENT REQUIRED PRODUCTION DEPLOYMENT

After the success of the prototype purchase predictor, it is planned to turn this into a production system. Whilst the prototype worked well, there are a few developments planned before the system is pressed into regular service:

- Consideration of how to deal with multiple editions of the same work. In most cases (as demonstrated by the academic's quote above) purchases should be for the latest edition of a work, so the system could use one of the related ISBN services (such as LibraryThing.com's ThingISBN [5]) to find these and add them in as alternative purchasing suggestions.
- The prototype concentrated on the acquisition of physical books, but the production system may need to expand its remit to include electronic articles, books and book chapters (e.g. material obtained from the Copyright Licence Agency [6]).
- Continuing work to speed up the predictor and reduce the load it imposes on the systems from which it extracts data. This will mean more caching, and crafting optimised database queries.

Further into the development it may be worthwhile considering adding some machine learning capabilities in the system. This would allow the system to adapt its future suggestions depending on previous purchasing patterns, allowing it to capture the subtle adjustments that library staff make to the mechanically derived lists it currently produces. It might also be possible to allow the academic departments to provide feedback directly into the purchase predictor, and allow them to use it as an additional conduit for updating their reading list information.

5 CONCLUSIONS

Leveraging the existing databases that the library has access to and combining them with third party data sources has allowed the purchase predictor system prototype to automate an otherwise long and tedious job for library staff. As it appears to be able to give suggestions broadly inline with those that the staff would make themselves after undertaking the previous manual procedure, the Library now has a reasonable level of confidence that the system is producing sensible purchasing suggestions.

In the long term, once the library has more experience with the use of the system and can make more comparisons between its suggestions with those of the library staff and academics, there is the option of allowing the system to make autonomous purchasing decisions, further reducing the load placed on library staff whilst ensuring the maximum benefit is gained from library budgets.

References

- [1] GP Brewerton and JP Knight. From local project to open source: a brief history of the Loughborough Online Reading List System (LORLS). VINE, 33:189195, 2003.
- [2] JP Knight, JL Cooper, and GP Brewerton. Re-developing the Loughborough Online Reading List System. Ariadne, 69:7, July 2012.
- [3] S Powers. Adding Ajax. O'Reilly, 2007.

[4] GNU General Public License, version 2, June 1991.

[5] LibraryThing.com. ThingISBN.

[6] Copyright Licensing Agency (CLA).