Predicting Book Use for Off-Site Storage[1]

Craig Silverstein and Stuart M. Shieber[2]

Abstract

We explore various methods for predicting library book use. Accurate prediction is invaluable when choosing titles to be stored in an off-site location. Previous researchers in this area concluded that previous use information provides by far the most reliable predictor of future use. Because of the computerization of library data, it is now possible not only to reproduce these earlier experiments with a more substantial data set, but also to compare their algorithms with more sophisticated decision methods. We have found that while previous use is indeed an excellent predictor of future use, it can be improved upon by combining previous use information with bibliographic information in a technique that can be customized for individual collections. This has immediate application for libraries that are short on storage space and wish to identify low-demand titles to move to remote storage. For instance, simulations show that the best prediction method we develop, when used as the off-site storage selection method for the Harvard College Library, would have generated only a fifth as many off-site accesses as compared to a method based on previous use.

1 Introduction

It is a commonplace that libraries never have enough room. For instance, Widener Library, the flagship of the Harvard College Library, comprises some 4.8 million volumes,[3] whereas the library building itself has space for only 3.5 million of these. Library systems must balance a desire to make titles generally accessible with the often prohibitive cost of increasing the capacity of on-site library locations. The solution to this dilemma was outlined over 250 years ago by Thomas Hollis, Harvard's second great library benefactor. (The first was John Harvard himself.) In a letter to the College Authorities of Harvard, Hollis proposed that "If you want roome for modern books, it is easy to remove the less usefull into a more remote place." [8] Former Harvard President Charles Eliot echoed the proposal in his well-known address to the American Library Association, recommending "the division of a library into books in use, and books not in use, with different storage methods for the two classes of books." [3]

The traditional solution to the overcrowding problem, then, is to move unused titles to a relatively distant off-site location, which can be built on less expensive land and can use efficient compact storage techniques. The Harvard College Library has taken exactly this approach using a depository library in Southborough, Massachusetts. At current rates of book acquisition, titles from Widener would not fill Harvard's depository for another 160 years. Thus, the depository can for all practical purposes solve the problem of inadequate storage. Ancillary benefits of a depository approach include climate control, enhanced security, and improved

preservation of fragile books. However, these benefits must be balanced against the great inconveniences libraries and library patrons experience with a depository system, notably the significant delay and cost inherent in retrieving volumes from the depository.

Given the necessity for off-site storage and its costs, it becomes important to choose those titles to be stored off-site carefully.[4] The dramatic impact of intelligent title choice can be shown graphically by comparing two *choice policies*, one in which the titles to be moved off-site are chosen *randomly*, and one in which titles are chosen *clairvoyantly* by looking into the simulated future and picking those books that are used the least in that time period. Figure 1 depicts a simulation of these two policies on a subsample of titles from Widener library. In the left portion of the figure, the percentage of titles to be stored off-site is varied along the horizontal axis. For each percentage, the choice policy is applied to choose titles to remove to off-site storage. We plot along the vertical axis the percentage of transactions in the simulated future that involve the removed titles. This percentage we call the *hit rate* for the policy. Of course, the lower the hit rate, the better a choice policy is.

One would expect that under a random choice policy, the hit rate should be the same as the off-site percentage. On the other hand, if we could clairvoyantly pick titles to be moved off site that we know would not be used, making our decisions with full knowledge of future use patterns, we would expect the hit rate would be reduced to 0 percent until most of the books are moved off-site. The figure verifies this expected performance; the hit rate is 0 percent for the clairvoyant

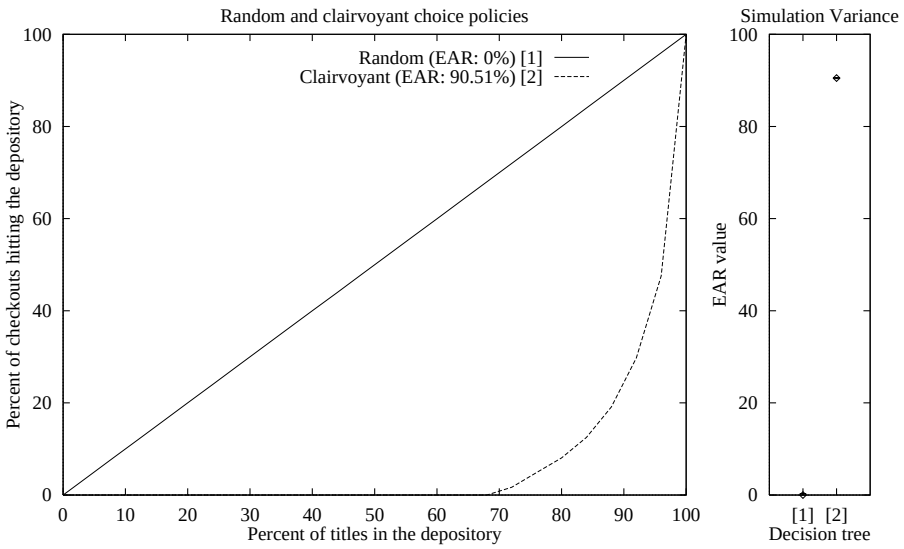

Figure 1: Performance of random and clairvoyant choice policies. These two provide expected extremes in terms of performance.

method up until about 70 percent of the titles are stored off-site. If we average the relative advantage of the clairvoyant method over the random method at all off-site percentages, we get the *Expected Advantage over Random* (EAR), which provides a measure of the quality of a choice policy. The EAR for the clairvoyant policy is around 90 percent, that is, we would expect (given no assumptions about how many titles to store off-site) that the clairvoyant policy engenders only one tenth as many hits off-site as the random policy.[5] The EAR values for the two choice policies (0 and 90 percent, respectively) are graphed in the right-hand portion of the figure. (This portion of the figure also shows error bars for the variance in the EAR value estimate, as described in Section 3.2.)

Although a clairvoyant policy is not implementable, the thought experiment

shows that there is tremendous room for improvement over random choice when deciding which books to put in a depository.

In this paper, we explore the design of choice policies by examining a large class of such policies, namely those that can be expressed as *decision trees* [14]. Essentially all past research on the topic of choice policies has worked with policies in this class, allowing us to replicate these results — albeit on a much larger scale. However, the availability of large databases of bibliographic and circulation information and the relatively more sophisticated computer resources now available allow us to go well beyond the simple decision trees previously considered. We can now examine trees orders of magnitude more complex, using subsamples of much larger scale and variety. In the process, we develop practical choice policies with an EAR of over 73 percent, compared to previously proposed policies with EARs of 45 to 60 percent.

2 Previous Research

In order to design a choice policy to minimize the expected use of titles stored off-site, we must have a model that allows predicting the use of individual titles. The two problems of predicting book use and designing choice policies are thus closely related (but see Section 5 for a discussion of some differences between the two problems).

Note that what is needed is a method of predicting the use of individual titles; it is not sufficient for our purposes merely to predict aggregate book use (though it is sufficient for other purposes, such as aiding resource allocation). A great deal of

previous research in predicting book use addresses the aggregate prediction problem [13, 10, 1, 5, 2, 6, 11], modeling future use distributions in toto as, for instance, mixtures of Poisson processes, perhaps incorporating a decay factor to model book aging. Such models are insufficient for our purposes for two reasons. First, we must select individual titles for removal off-site; aggregate statistics provide no aid in determining which books will fall in the tails of the distribution, merely predicting what the distribution will look like. Second, aggregate models must somehow extrapolate past use as a predictor of future use. Thus, they presuppose exactly the kind of model that we are interested in developing. However, because the information they must deliver is much coarser, there is little or no motivation to subdivide the books into subclasses for separate prediction; this would only be useful if the combination of the aggregate predictions for the two classes were more accurate than the aggregate prediction for the union of the two classes. For this reason, aggregate prediction methods can (and should) rely on relatively simple models of prediction, essentially just past use. The sophistication in the models is in the distribution modeling method or the combination with economic models, not in the prediction criteria.

There is, however, a body of previous research on predicting library book use that is applicable to designing choice policies. Almost all researchers agree that some direct measure of past use is the most significant predictor of future use. Fussler and Simon, in their seminal 1969 study, concluded that past use is the only good indicator of future use [4]. Slote, in his study of 1971, agreed: "The most recent

use pattern as reflected by the current circulation is a strong and valid predictor of the future use [of books].... The most recent shelf-time period should be the sole criterion for weeding and for identifying useful core collections" [15, p. 34]. Only one study, by Jain [15], claims to find a single predictor that does better, but as Slote notes, Jain does not really offer any evidence that his predictor is better than past use.

The value of past use information is limited, however, by the fact that many titles in a research library are never checked out at all, and hence have identical past use history. In 1975, for instance, researchers at the University of Pittsburgh determined that 40 percent of all titles purchased by the university had never been checked out [9]. Data from Harvard's computerized transaction database show that these results transfer to Widener library: 40 percent of all titles in Widener have not been checked out in the last 20 years, the period for which computerized records exist.

An obvious method for distinguishing among these low-use titles is to take into account bibliographic information when trying to predict future book use. Fussler and Simon did so, examining combinations of decision criteria for predicting book use. For instance, they ranked titles that had never been checked out based on their date of acquisition by the library. In addition, they analyzed titles in different LC classes separately, so LC class was also an implicit decision criterion in their study. These two criteria were used in addition to past use statistics; when Fussler and Simon disallowed past use statistics as criteria, they needed to combine even more

criteria to obtain an adequate choice policy.

Before the advent of computerized transaction systems, research into these more complicated decision schemes was limited by the computational difficulty of managing all the necessary information about library books. Possible prediction methods had to be arrived at through *ad hoc* methods, and only small amounts of data could be gathered to evaluate their efficacy. Fussler and Simon's study, probably the largest, examined a total of only 1642 titles — all of them books — generating 1601 transactions. Their titles were not uniformly distributed throughout the library system, but were instead concentrated in two subject areas, Economics and Teutonic Literature. In addition, the researchers found it difficult to estimate past use information, since the sheet of paper containing book use information, stored in the rear of each volume, was replaced after a few dozen entries. The computerization of library records, combined with computer algorithms that can be automatically trained to predict book use, makes possible a more complete examination of algorithms for predicting library book use, which we describe below.

2.1 Replicating previous results

With the availability of computerized databases of bibliographic and circulation information and more sophisticated computer resources, we can reproduce these previous studies using a much larger data set in a more exhaustive evaluation. The decision criteria that we examine are shown in Table 1. These criteria include all those examined by Fussler and Simon with the exception of acquisition date,

which was not available for the Widener data[6]. In addition, we consider CHECKOUT HISTORY and COUNTRY OF PUBLICATION, which Fussler and Simon did not. The first two of these criteria — CHECKOUT HISTORY, measuring the number of past uses, and LAST USE, measuring the time since the last use — we refer to as *past use* criteria, as they rely on the book's circulation behavior in the past. In the experiments reported on here, we use circulation records from July 1975 to June 1984 as the "past" for use by these criteria. Data from July 1984 to June 1993 are not available to the past use statistics and are used to analyze decision trees. (See Section 3 for a fuller description of the methodology of evaluating decision trees.)

In Figure 2 we compare the performance of various decision criteria on an 80,000 title random subsample of books in Widener library. For comparison purposes, the figure also includes the benchmark policies based on random and clairvoyant choice.

Consistent with previous studies, the past use criteria definitively outperform other criteria. The best of these criteria, CHECKOUT HISTORY, has an expected advantage over the random policy of 58.9 percent. However, when few titles need to be stored off-site, the advantage of past use criteria is not so clear cut. Figure 3, which details the performance of the criteria at small off-site percentages, shows that LANGUAGE is the best performer in this range, and both LANGUAGE and LC CLASS outperform the past use criteria. Previous studies may not have noted this behavior since it only holds when less than about 18 percent of all titles need to be moved off-site; Fussler and Simon's study, for instance, only tested their criteria with simulations of 25, 50, and 75 percent of titles stored off-site.

Criterion	*Description* *Example values [number of attested values]*
CHECKOUT HISTORY	Number of times the book circulated in the past. 0 times, 1 time, 9 times, 1898 times [90 values]
LAST USE	Number of months since the last use in the past. 0 months, 1 month, 108 months, never used [110 values]
LC CLASS	Alphabetic prefix of the Library of Congress call number. Harvard University keeps some titles under an older classification scheme. Such titles are given an "LC class" by prefixing the Widener prefix with "WID". A, PQ, WID ECON [486 values]
PUBLICATION DATE	Date of publication (not acquisition) of the book. 1789, 1900, 1986 [357 values]
LANGUAGE	Language in which book is written. English, Swahili, Achinese [127 values]
COUNTRY	Country in which the book was published, following the Library of Congress specification, in which states of the US and certain other sub-national units are classified as countries. Australia, West Germany, Massachusetts [276 values]

Table 1: Criteria considered in models of predicting book use. The number of values evidenced in an 80,000 title sample is provided in brackets in the second column.

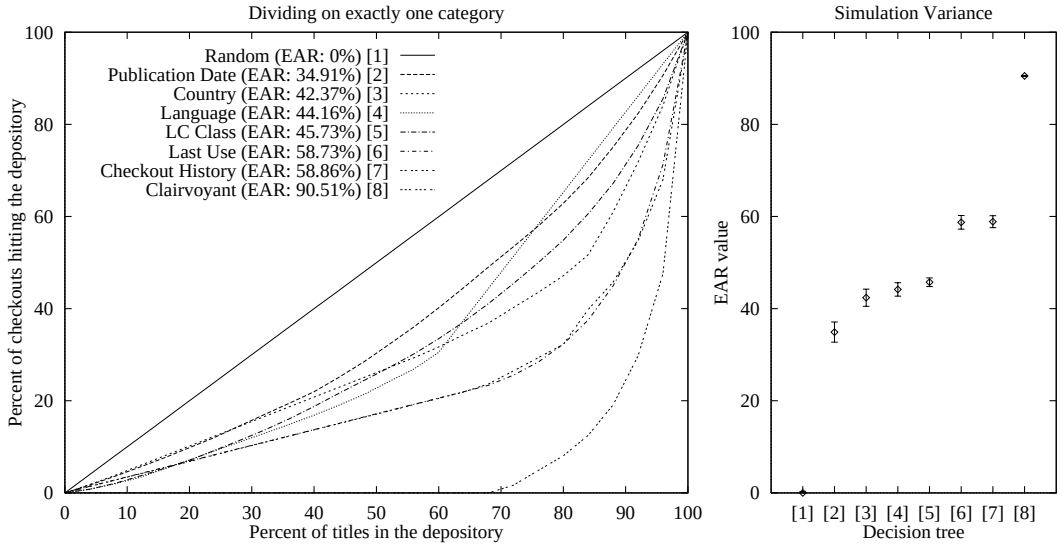

Figure 2: Performance of single criterion choice policies. The two past use criteria perform the best overall.

Using the same 80,000 title sample, we can evaluate approximations to the various multi-criterion policies that Fussler and Simon developed, and compare them to the LAST USE criterion, the best-performing single criterion. The comparative results are shown in Figure 4. The "no use records" policy uses LANGUAGE and PUBLICATION DATE to rank titles, while the "use records" policy uses LAST USE and PUBLICATION DATE. Fussler and Simon did not explicitly consider LC CLASS as a predictive criterion, but did treat separately the two LC classes they studied, so we also look at the performance of the two policies when augmented with the LC CLASS criterion.

Of the first two variants, the one taking advantage of past use information is

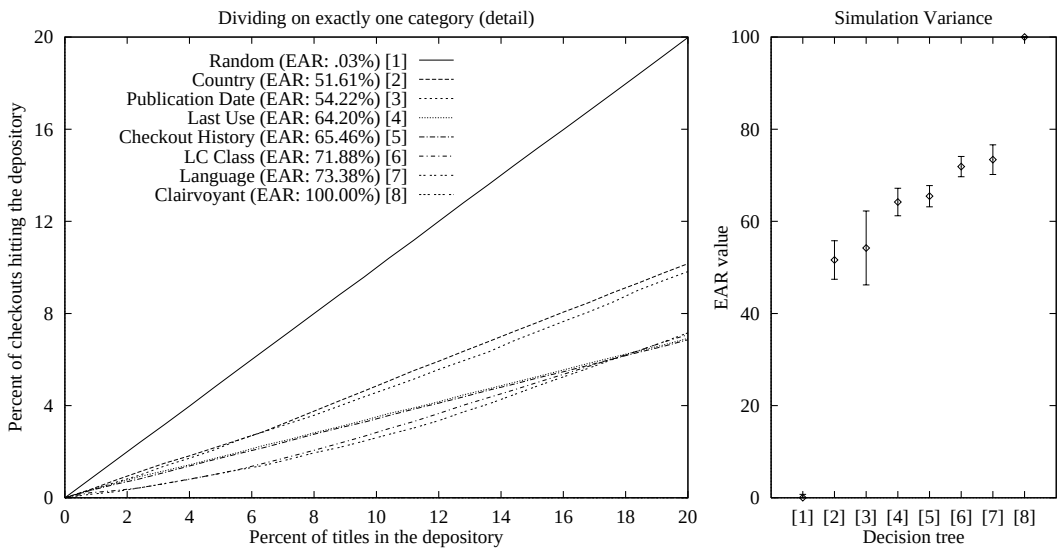

Figure 3: Detail of Figure 2 for small off-site percentages. In this range LANGUAGE and LC CLASS outperform the past use criteria. The clairvoyant line cannot be seen because it lies along the horizontal axis.

again better overall, but when few titles need to be put in the depository, it performs worse than the method based purely on bibliographic information. This makes sense, since in this low range all titles being considered will have never been checked out, and the LAST USE policy will be picking titles basically at random. Surprisingly, using LC CLASS as an extra criterion can degrade performance: Adding LC CLASS to the "no use records" policies reduces the EAR from 52 to 47 percent. It is possible that adding another criterion makes the algorithm prone to *overtraining*. (See Section 4.3.)

In summary, past use statistics are the best single criterion for predicting book use, although contra previous studies, other criteria dominate when small percentages of books (less than 15 percent) are stored off-site. The addition of extra criteria to past use can, to an extent, further improve predictive power, but care must be taken, as degradation can also result.

3 The Methodology of Decision Trees

The multi-criterion algorithms of Fussler and Simon can be seen as variants of a general class of methods based on *decision trees*. A decision tree is a hierarchical structure for classifying objects, composed of *nodes* that correspond to primitive classification decisions. For the task at hand, the objects to be classified are titles and we wish to classify them in such a way that the classes are maximally predictive of their future use. The primitive classification decisions are simply the criteria that are available for classifying titles, as listed in Table 1.

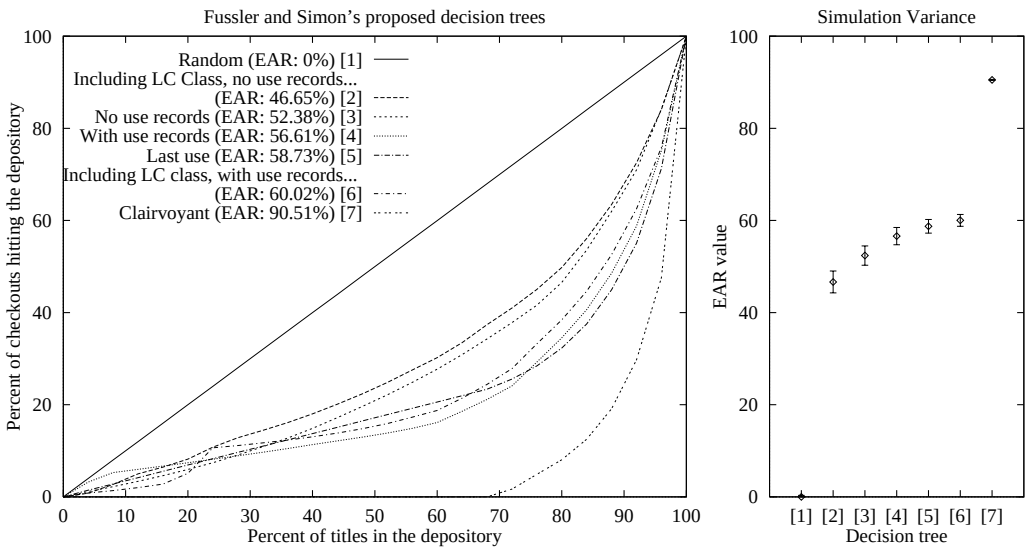

Figure 4: Performance of choice policies recommended by Fussler and Simon [4]. Policies using use records outperform those that do not in general, but not when less than 20 percent of books are in the depository. Adding LC CLASS does not always improve performance, perhaps because of overtraining.

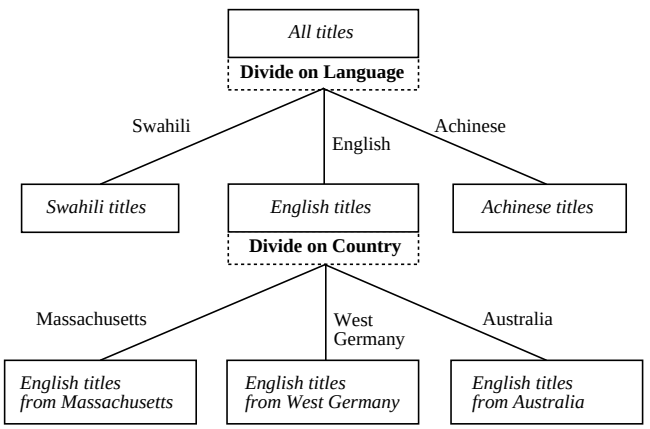

Figure 5: A simple decision tree.

At the top of a decision tree (called, perversely, the *root*; decision trees grow down rather than up) is a node that specifies the main dividing criterion for subclassifying the titles. The dividing criterion might be, for instance, LANGUAGE. For each value of this criterion — English, Swahili, Achinese, etc. — the node has a child node, which can be thought of as classifying further all the titles with the given language value. Associated with each node, in addition to a dividing criterion, is a set of titles. The root node contains all the titles, while child nodes contain those titles they inherit from their parent. In our example, the ENGLISH child node inherits from its parent all titles written in English, while the SWAHILI node inherits titles in Swahili, and so on.

Each of these nodes in turn can have a dividing criterion and children of its own. In this way, the set of titles can be subclassified into finer and finer subgroups, where the nodes at the bottom of the tree, the *leaves*, constitute an exhaustive classification of all of the titles into disjoint classes. Figure 5 shows a simple decision tree. Titles are divided first on the basis of the language they are written in. Those written in English are further subdivided on the basis of their country of publication. (For purposes of exposition, we limit attention in the figure to a small subset of the possible values for these criteria.)

3.1 Defining and evaluating choice policies

A decision tree can be used as the basis of a choice policy by ordering the leaf nodes according to which classifications are expected to have the lowest hit rate. We

calculate this expected hit rate for each node using data from 1984–1993. (Recall that data from 1975–1984 were used by the past use statistics.) That is, each node is assigned a value based on the number of times titles in that node were checked out in the period 1984–1993. We can, in this case, think of the data from 1984–1993 as the "recent past" and data used by past use statistics as stemming from the "distant past."

This ordering of nodes induces a corresponding ordering on the associated sets of titles. When we need to pick titles to place off-site, we start taking titles (in arbitrary order) from the lowest-ranked node's classification, moving on to higher-ranking nodes as the earlier nodes are emptied. If use patterns from the recent past, which we used in our ordering, hold into the future, the ordering we generate will be the best one.

This procedure is adequate for creating and using decision trees, but we need some more data in order to test their efficacy. One way to do this is to garner some "future" data and, as we move titles to the depository, track how many times the titles are checked out in the "future." This technique allows us to calculate the hit rate at different percentages of off-site storage. In this way we can generate curves such as those in the previous figures.

This routine can be improved upon by using a different data set to test the decision tree, which we call the *testing set* to distinguish it from the *training set* used to construct the structure and ordering of nodes in the decision tree. This way we are sure that we are evaluating the predictive power of the way the tree divides

titles in general as opposed to its predictive power on the specific training set. We would use testing set circulation data from the "recent past" to calculate past use statistics; this is necessary so that we can determine the appropriate node for each title of the testing set. Then we could use the data from the "future" to evaluate how well the tree predicts future use.

Unfortunately, we have no data from the future. However, since the data in the testing set are unrelated to the data in the training set, it is acceptable to move the entire time frame for the testing set backwards. Therefore we can use the "distant past" for past use information and the "recent past" to simulate the future. This technique will fail only if patterns of use change rapidly enough to make predictions based on the "recent past" (1984–1993) invalid in the future. We do not expect this to be the case.

A few examples may clear up the process. The random choice policy is defined by a decision tree with a single node. Since no dividing goes on, all titles in the training set are placed in that one node as a single class. That node is the only leaf node, so that the sorting of the leaf nodes according to their recent past use is trivial. Next the node is cleared and replaced by titles in the testing set; again since there is only one node all the titles are placed in it. These titles are then chosen in an arbitrary order to be moved to the depository. Since the books are chosen arbitrarily from the single group, hit rate goes up linearly with off-site percentage.

The single criterion choice policies of Figures 2 and 3 correspond to decision trees of a single level, with a root node that divides on the given criterion. For example,

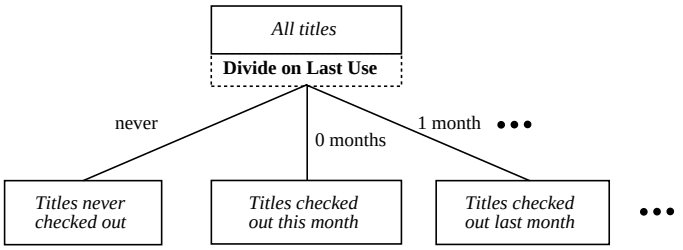

Figure 6: A single-criterion decision tree that divides on the LAST USE criterion..

consider the LAST USE single criterion choice policy. The corresponding decision tree is depicted in Figure 6. The root node is the only non-leaf node of the tree. The leaf nodes are sorted based on their performance in the "recent past" (undoubtedly with more recent last use nodes sorted ahead of less recent ones). The nodes are then cleared of titles and replaced with titles from the testing set. The "distant past" behavior of the testing set titles is used to divide the titles according to the LAST USE dividing criterion. The titles from the testing set are then selected for off-site storage according to the sorting order previously determined. The "recent past" use data for the testing set are then used to determine the hit rate as a function of off-site percentage.

The "no use records" Fussler and Simon policy of Figure 4 corresponds to a decision tree where the root node divides on LANGUAGE and the children of the root node divide on PUBLICATION DATE. Figure 7 displays this decision tree.

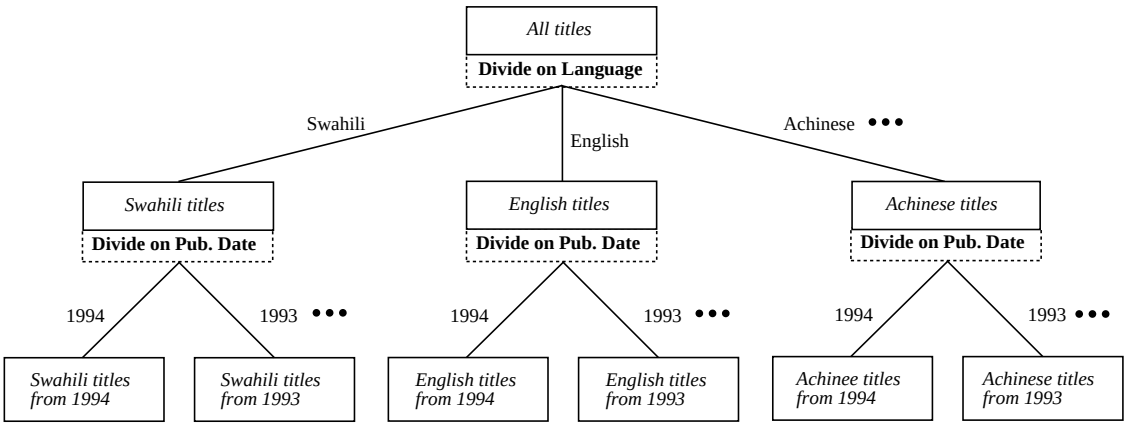

Figure 7: One of Fussler and Simon's choice policies shown as a decision tree.

3.2 Sample size

At the time of this study, 2.2 million of the estimated 3.8 million titles belonging to Widener have been cataloged in Harvard University's on-line library computer system, having generated a total of 6 million transactions since July 1975. A campaign is in progress to computerize the rest of the titles, most of which have not generated a single transaction. Because of computational limitations, we do not divide the entire collection into two data sets of 1.1 million titles each. Instead, we make the training and testing sets somewhat smaller, approximately 80,000 titles.

In subsampling the data, it is important to ascertain that a sufficiently large subsample is being used. Figures 8 and 9 show the performance of a simple LAST USE decision tree on subsamples varying in size from 5,000 to 80,000 titles. Notice that the performance of the decision tree converges rapidly as the sample size increases, indicating that the results of our experiments, which use data sets of 80,000 titles,

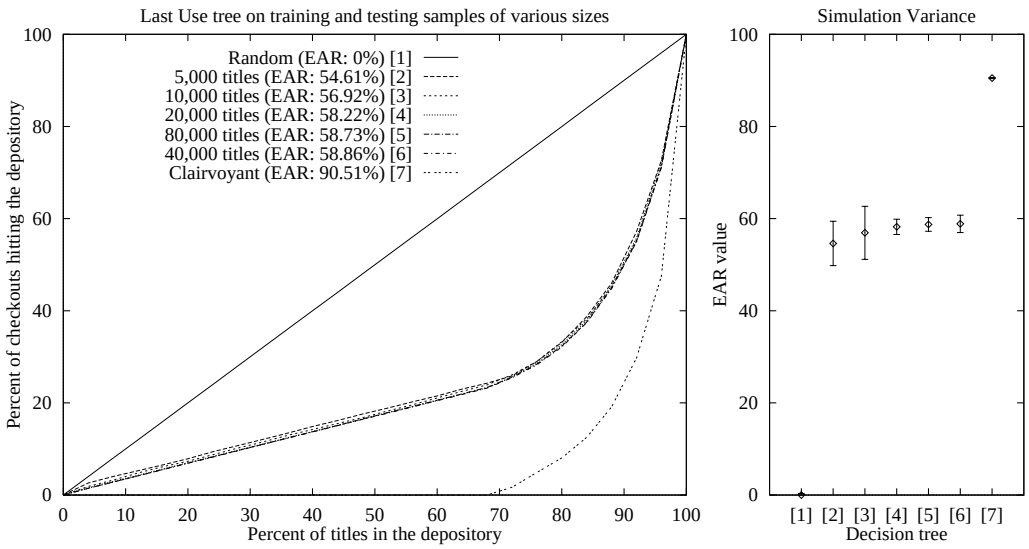

Figure 8: Performance of LAST USE criterion with subsamples of various sizes. As sample size increases, performance improves slightly as well, quickly converging to a standard value.

should be applicable to larger and smaller data sets as well. The same trend holds for other decision trees, including the much more complicated ones discussed in future sections. Thus, a sample size of 80,000 titles seems sufficient.

As another verification of the reliability of the EAR figures that we calculate through simulations, we calculate the maximum error for the EAR value of each choice policy due to subsampling. Each choice policy is tested on eight different test sets. In the left-hand portions of the figures, a set of averaged curves is shown, one for each choice policy. In the right-hand portion of the figure, the mean EAR value is shown with error bars bracketing the 95 percent confidence interval. The right-hand policies are keyed to the left-hand legend with the bracketed numbers.

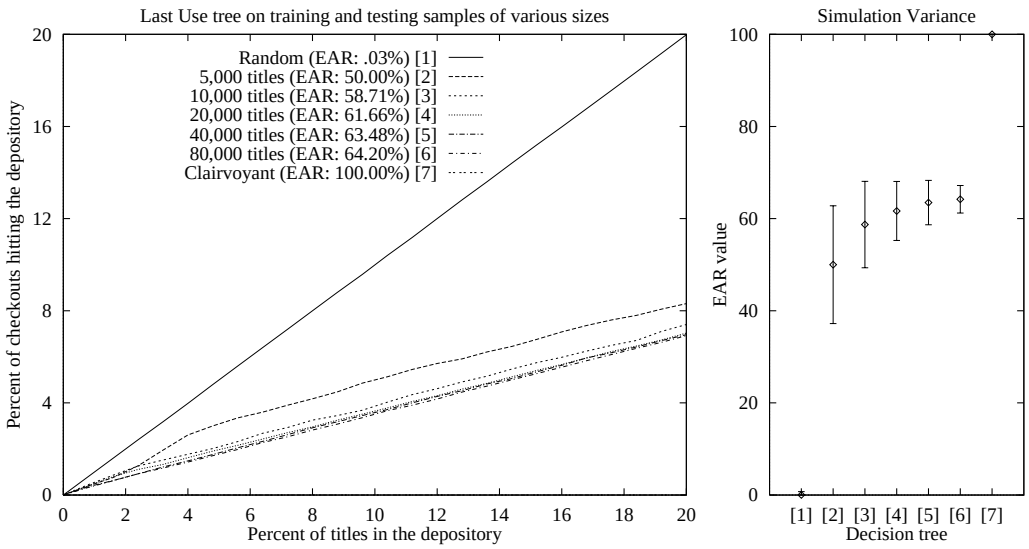

Figure 9: Performance of LAST USE criterion with subsamples of various sizes, detail.

In general, the variances are quite small and serve to confirm the significance of the differences among the choice policies.

4 Designing Decision Trees

Once we characterize previous algorithms for predicting book use — from simple one-criterion tests to the more complicated tests developed by Fussler and Simon — as decision trees of one or two levels, it seems natural to look at even larger decision trees. After all, it surely cannot hurt to consider as much information about a title as possible before deciding whether to move it off-site.

Once a node in a decision tree divides on a given criterion, it makes no sense to

divide again lower in the tree on that same criterion. Therefore, if we can consider up to n criteria, a tree can include at most n levels. Of course, we may choose to make a decision tree that is not so deep. Trees of maximum depth we call *maximal* trees. Since the present study examines six different criteria (Table 1), maximal trees have six levels between the root and each leaf node. In order to maximize the amount of information used in making storage choices, we should consider maximal decision trees.

Indeed, the question arises as to why previous researchers did not examine these more complicated methods themselves. The reason is undoubtedly one of impracticality: the largest decision tree we consider has 140,000 nodes, making it impossible to analyze without a computer. Even with the use of computers, computational limitations remain. Instead of creating a decision tree, which considers only a single classification criterion at a time, it might be better to calculate a regression analysis on the various criteria, determining not only how each criterion contributes to future book use but also how combinations of criteria contribute. Unfortunately, such analyses on data sets of greater than 5,000 titles, even when calculated using only a limited subset of the possible criteria, proved to be impossible given current technology. Nevertheless, it is certainly possible to evaluate maximal decision trees using the current technology.

4.1 Randomly selected maximal decision trees

All maximal trees differ only in the order of the dividing criteria; their leaf nodes classify the titles into the same disjoint classes. That is, the maximal trees have the same leaves but in a different order. Since our evaluation technique reorders the leaf nodes (based on predicted hit rate), all maximal trees are theoretically the same for the purposes of evaluation. It should not matter which we choose to evaluate.

The only exception to the equivalence of maximal trees would arise if the observed hit rate for two nodes is identical. In this case, some method for "breaking the tie" must be instituted so as to fully order the nodes. Unfortunately, such ties are quite common. For example, one maximal tree for an 80,000 title sample has 46,000 leaf nodes but only past use 247 values for those nodes. There tend to be many ties because each leaf contains only a few titles. In contrast, a simple decision tree based only on LAST USE might have 110 leaf nodes with 93 different values. Nodes with the same value look the same to any ordering algorithm, so a tie-breaking method must be invoked.

The hierarchical structure of the decision tree turns out to be useful here. If two nodes are tied in the number of checkouts of their constituent titles, we can compare the nodes' parents instead. The parents have less rigorous dividing criteria, and therefore contain more titles; the more titles a node has, the less likely its average hit rate is repeated in some other node. If another tie does arise, the remedy can be repeated, leading to examination of grandparents, and so forth. Only rarely will this technique fail to differentiate between two nodes, forcing us to pick one over the

other arbitrarily. Note that an arbitrary decision here is not necessarily appropriate: it may be that one node is superior to the other in fact, but our sample size is too small to let us determine it. By using the parent nodes to perform the ordering, we can effectively increase the sample size at the cost of some specificity.

Thus, for large trees, the ordering of the decision criteria becomes important in determining the breaking of ties. (Order of division is also important when we smooth the maximal trees to eliminate problems of overtraining, as discussed in Section 4.3.) This can lead to varying performance among the different maximal trees.

In order to gauge the quality of maximal trees in general, then, we normalize for this effect by selecting a different maximal tree randomly for each evaluation; each node is randomly assigned a dividing category from the set of legitimate categories remaining to it. This randomness explains the large variance in the exhibited performance for maximal trees.

Figure 10 presents a comparison of such random maximal trees against the various trees proposed by Fussler and Simon. Surprisingly, the maximal tree is not the definitive top performer. For instance, Fussler and Simon's past use tree differentiated by LC class — which divides only on LC CLASS, LAST USE, and PUBLICATION DATE — is significantly simpler than the maximal tree, but performs almost as well, particularly when few titles need to be put in the depository. This does not mean, however, that maximal trees are inherently flawed; in fact, due to the large variance it is hard to tell exactly how well maximal trees perform. We need a way of choosing

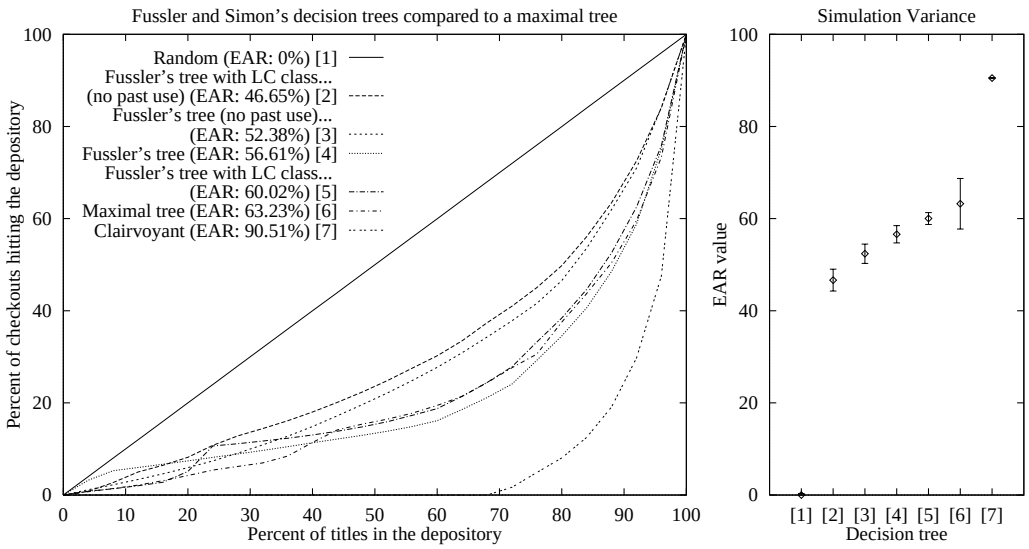

Figure 10: Performance of a maximal decision tree compared with choice policies recommended by Fussler and Simon [4]. The maximal tree does not perform much better than the others despite its greater complexity.

the order of dividing on maximal trees in order to reduce the variance, preferably improving mean performance at the same time.

4.2 ID3-ordered decision trees

It is possible to use an algorithm called ID3, developed by Quinlan [14], to find a good dividing order. For each node, the ID3 algorithm uses a heuristic to calculate the *information gain* inherent in dividing on different criteria and picks the criterion with the greatest gain. Information gain is highest when the output values of the children of a node are as different as possible. For instance, suppose the titles in a node are checked out on average 3.5 times in the "recent past" — the time period

used to order the nodes. If we divide the node and create two children, one with an average checkout of 0 and the other with an average of 7, we have gained a lot of information because it is easy for us to decide which node to rank higher. On the other hand, if the children have average checkouts of 3.4 and 3.6, we have gained less information. For a mathematical description of how the ID3 algorithm makes its decision for dividing criterion, see Appendix A.

The ID3 algorithm is not guaranteed to give improved performance. However, we see in Figure 11 that an ID3-ordered tree performs better than a maximal tree on average. The mean EAR value for the ID3-ordered tree lies barely within the 95 percent confidence interval for maximal trees, showing that the ID3 ordering is better than the vast majority of orderings for maximal trees. Equally important, the ID3-ordered tree has a much smaller variance.

4.3 Overtraining and smoothing

In addition to being subject to the tie-breaking problem described in the previous section, large decision trees are prone to *overtraining*: As the tree classifies titles into finer and finer classes, the ordering of the leaf nodes tailors itself to idiosyncrasies of the data set it is training on. Instead of capturing trends relating criteria to future use, the tree captures information specific to the particular titles they are trained on. As an extreme case, consider a tree so large that it has one leaf node for every title. Then the nodes would be ordered based on the use of each title in the training set, capturing a lot of information about the title itself but none about the criteria

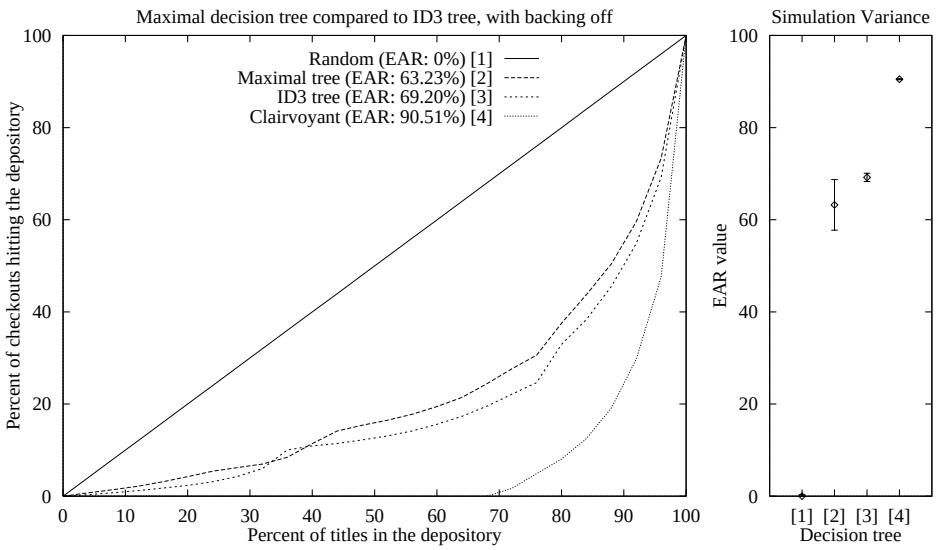

Figure 11: Performance of ID3-ordered tree and a maximal tree. The ID3-ordered tree outperforms the maximal tree with its dividing order picked at random. In addition, it has a smaller variance, making it easier to analyze its efficacy.

appropriate for the title. Unfortunately, this specialized information is useless once the titles are replaced with titles from the testing set. One way to reveal that a tree is overtrained, then, is to test it on the training set. Since overtrained trees are optimized for the data set they were trained on, they do much better when tested on their training set than when tested on a separate testing set.

This problem is inherent in the size of the tree and cannot be solved by reordering the nodes. In fact, we see in Figure 12 that the ID3-ordered tree suffers greatly from overtraining. Performance of the tree when tested on the training set approaches that of the clairvoyant policy, showing that the training process was quite successful in tuning the ordering to the training data. The far inferior performance on separate testing data shows that much of this training was spurious.

Overtraining can be eliminated by *smoothing* the tree to remove excess leaf nodes. We smooth by combining many small nodes into a larger node. Instead of many unreliable, "bumpy" data points from many small nodes, we have one "smooth" data point that averages the behavior of the small nodes. Good smoothing methods leave many nodes where they are needed to make fine distinctions between titles but eliminate nodes that contribute to overtraining without making any real contribution to categorization accuracy.

We explore the use of two smoothing methods on the maximal decision tree. The first uses a heuristic to find places where the tree is more developed than the data warrant. The heuristic looks at the variance of the hit rates for titles in each node. Nodes with small variances are themselves good predictors of book use and do not

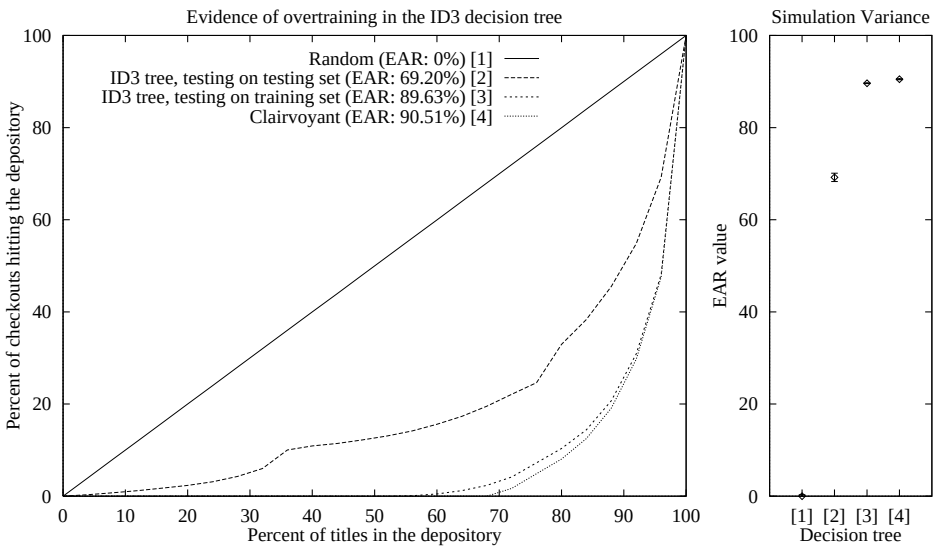

Figure 12: Performance of an ID3-ordered decision tree when tested on the training set and when tested on a separate testing set. The far superior performance of the tree in the former case is evidence of overtraining.

need to be divided further. The algorithm looks for such nodes, choosing the node whose variance is the smallest with respect to the average variance of its children. It then makes that node a leaf node, deleting its children and further descendents from the tree. The process is repeated, creating a succession of smaller and smaller trees. Each tree is evaluated and the tree with the best performance is chosen as the smoothed tree. Because of the way children are removed, we refer to this method of smoothing as *pruning*.

The second method of smoothing works in the opposite direction: Instead of pruning the children of a given node, it folds a given node into its parent. We refer to this method as *backing off*. We back off a node if we doubt the reliability of its estimate of the future hit rate for that node. We determine reliability by looking at the number of circulation events that the titles classified by the node account for. In particular, we define a node's *size* as the number of titles in the node plus the number of past checkouts for all titles in the node. Nodes with small size are unreliable because their scant use history increases the variance of their future use information. We ignore small nodes, considering their much larger parents in their stead. As in the pruning method, a succession of trees is created, each formed using a different maximal size for backing off. Again the tree with the best performance is chosen as the smoothed tree.

Both smoothing algorithms generate a series of trees and require us to judge their performances. What data set can be used to make this judgment? We cannot use the training set, because the whole point of smoothing is to alleviate the tree's

dependence on the training set. We cannot use the testing set for the same reason we cannot train using the testing set: it would constitute cheating and cause the testing procedure to underestimate the true hit rate. We must instead use a third data set, which we will call the *smoothing* data set. To do this, we divide the training data set in two, using half of it as the new training set and the other half as the smoothing set.[7]

We compare ID3-ordered trees produced by the two smoothing methods with the unsmoothed tree in Figure 13. The backed-off tree, with an EAR of 73.1 percent, performs the best, while the pruned tree performs worse than the unpruned tree, leading us to question the efficacy of the pruning heuristic. As shown in Figure 14, the backed-off, ID3-ordered tree performs consistently better than trees developed by Fussler and Simon and other researchers. The improvement is particularly striking when 20–40 percent of titles need to be moved off-site, a reasonable range for many research collections. In all ranges, however, the backed-off, ID3-ordered tree is the best decision tree we have studied.

5 Problems

Though the ID3-ordered tree performs well, it comes far from approaching the performance of the clairvoyant tree, which provides an upper bound on the best performance possible. Part of the problem may be in the ID3 algorithm itself. While an adequate heuristic, it has been surpassed in recent years by algorithms that can divide on linear combinations of the data and otherwise improve the criterion de-

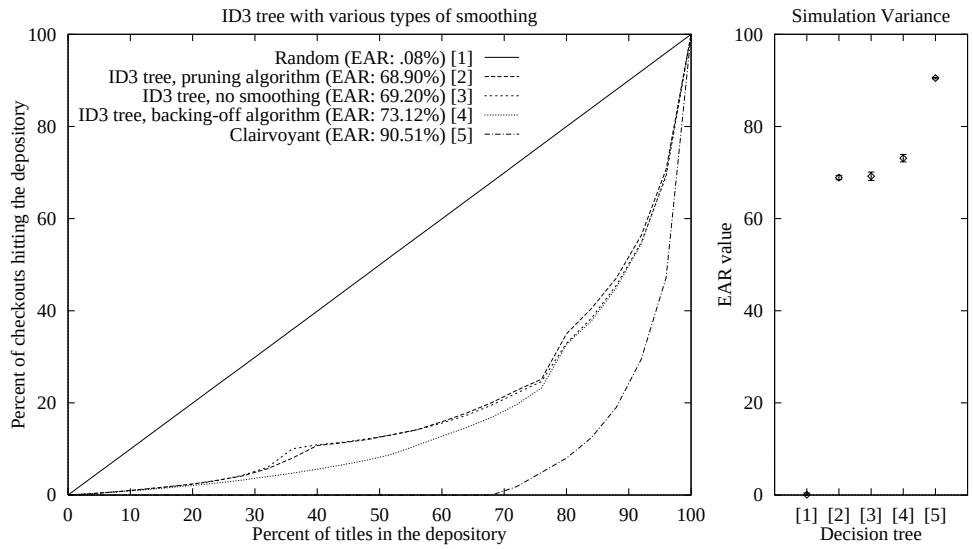

Figure 13: Effect of smoothing on performance of ID3-ordered trees. The right type of smoothing can dramatically improve performance.

cision procedure. Unfortunately, these more complicated algorithms do not lend themselves to the large number of criterion values found in the library data set, which make their computational cost prohibitive. If a more modern algorithm can be tailored to the library data, however, it may give improved performance.

Another problem may be merely that not enough criteria are being considered. With good smoothing methods, it is possible to include many more than the six criteria we considered while not overtraining the decision tree. The problem remains of finding other predictive criteria. Preliminary examination of the type of a title — monograph, serial, map, etc. — indicates that this statistic does not improve the accuracy of prediction. Other criteria available in the Harvard bibliographic database, examining the author's name, for instance, or whether the title includes

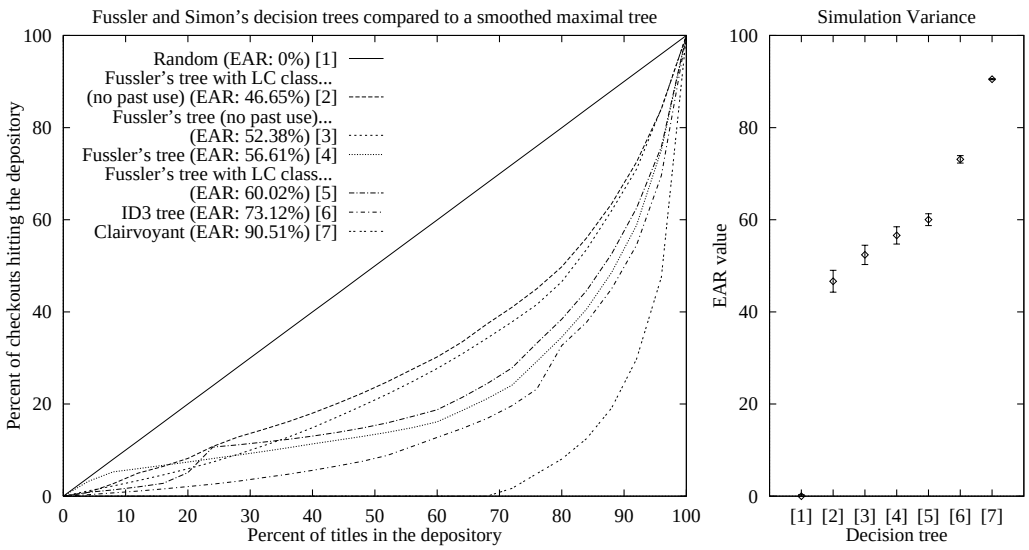

Figure 14: Performance of a backed-off, ID3-ordered tree compared to trees recommended by Fussler and Simon [4]. The smoothed, ID3-ordered tree is clearly superior to the simpler trees.

illustrations, are even less likely to improve performance.

Even if decision tree methods could be refined to give near-clairvoyant performance, there is still considerable debate over whether such a method should be used at all. Two main schools of thought have developed as to how to pick titles to move off-site. One, which may be titled the "What readers *should* want" school, proposes that experts in various fields pick those titles in their field that are the least "worthy" and send them to remote storage. Proponents of this view hope that because of the work involved in recalling a title from the depository, casual students of a topic will be guided to the higher quality titles in the main collection (see, e.g., [18, page 201]). The other school of thought, the "What readers *do* want" school, believes the library should try to minimize use of the depository by predicting patterns of future use. In this way, titles that will not be used often in the future can safely be placed in the depository. We have followed the latter approach, as in previous studies of library book use [4, 16, 17].

Finally, there are methodological questions concerning the study itself. Like most such studies, we do not include in-house browsing in our analysis; we must be satisfied by previous reports [4, 12, 7] that concludes the frequency of in-house use is highly correlated with the frequency of checkouts, though these conclusiuons are admittedly controversial [5, 6]. Furthermore, we do not study all the titles in Widener, since some 1.5 million titles are not yet in the computer system.[8] The results we have obtained thus apply not to Widener as a whole but to some "sub-Widener" that excludes many relatively unpopular titles. However, the relative

comparisons are still valid, assuming that one decision scheme would not benefit inordinately from the non-computerized titles. This seems probable. Once the ongoing effort to computerize all titles has been completed, it should be possible to tailor the decision-making policy to the true population of Widener.

One way to ameliorate many of the problems noted here is to combine an automated procedure with manual oversight. This can be done in a variety of ways. A manual pre-process stage could specify individual titles or classes of titles (such as reference materials or other titles used frequently in-house) to be exempted from consideration by the automated procedure. A manual post-process stage could combine a computerized method with expert review. For instance, a decision algorithm could be made to list more titles than need to be moved, and experts could pick the requisite number of books from the candidate list. In either case, a good prediction algorithm can ease the burden of deciding on titles to move to remote storage when a main collection becomes too crowded, even if it does not eliminate the burden entirely.

6 Conclusions

In this study, we took advantage of the availability of large databases of bibliographic and circulation information to replicate and extend earlier results on the prediction of library book use as it applies to choice policies for off-site storage.

We follow most previous studies by endorsing past use as the best single predictor of future use. We do so, however, with reservations. When a large percentage of a

library's collection needs to be held off-site, the best criterion is undoubtedly time since last use. When less than 15 percent of a collection needs to be moved, however, past use is less useful. This is because up to 40 percent of a collection may have never been checked out, and past use statistics are unable to distinguish among the books in this subpopulation. Instead, LANGUAGE OF PUBLICATION or LC CLASS seem to be the best criteria when few titles need to be put in the depository.

It is possible to combine the best of all worlds by using more than one criterion to predict future book use. The logical extension of this is to use *all* the criteria available in our prediction rules. Unfortunately, this causes decision trees to be too large for the data set, causing several problems. One can be solved by picking the nodes of the tree carefully; computational heuristics such as ID3 can be used to try to pick the best ordering. The rest of the problems can be solved by smoothing, which shrinks the decision tree in places the extra granularity is not needed. The smoothed, ID3-ordered, maximal decision tree convincingly outperforms any single-criterion decision tree, and is the best method we have tested for predicting book use.

By way of illustration, if the Harvard College Library had implemented a LAST USE policy to choose which 20 percent of its collection to move to the depository in 1985, they would have had to retrieve volumes from the depository about 34,000 times per year. If they had, instead, used the smoothed, ID3-ordered, maximal tree, there would have been less than a fifth as many, only 6,200 hits per year. In comparison, a random choice policy would have resulted in 60,000 hits per year,

while a clairvoyant policy would have garnered zero hits.

References

[1] Quentin Burrell. A simple stochastic model for library loans. *Journal of Documentation*, 36(2):115–132, June 1980.

[2] Quentin Burrell. A simple empirical method for predicting library circulations. *Journal of Documentation*, 44(4):302–314, December 1988.

[3] Charles William Eliot. The division of a library into books in use, and books not in use, with different storage methods for the two classes of books. *Collection Management*, 2(1):73–82, Spring 1902 [1978].

[4] Herman H. Fussler and Julian L. Simon. *Patterns in the Use of Books in Large Research Libraries*. The University Chicago Press, 1969.

[5] Robert M. Hayes. The distribution of use of library materials: Analysis of data from the university of pittsburgh. *Library Research*, 3(3):215–260, 1981.

[6] Robert M. Hayes. Measurement of use and resulting access allocation decisions. *Library and Information Science Research*, 14(4):361–377, 1992.

[7] Anthony Hindle and Michael K. Buckland. In-library book usage in relation to circulation. *Collection Management*, 2(4):265–277, 1978.

[8] Thomas Hollis. *Letter to the College Authorities at Harvard*, 1725. Cited by J.A. Urquhart and N. C. Urquhart, *Relegation and Stock Control in Libraries*, Oriel Press Ltd., Northumberland, England, 1976.

[9] Allen Kent, Jacob Cohen, K. Leon Montgomery, James G. Williams, Stephen Bulick, Roger R. Flynn, William N. Sabor, and Una Mansfield. *Use of Library Materials: The University of Pittsburgh Study*. Marcel Dekker, Inc., New York, NY, 1979.

[10] Gerald J. Lazorick. Patterns of book use using the negative binomial distribution. *Library Research*, 1:171–188, 1979.

[11] Hur-Li Lee. The library space problem, future demand, and collection control. *Library Resources and Technical Services*, 37(2):147–166, 1993.

[12] William E. McGrath. Correlating the subjects of books taken out of and books used within an open-stack library. *College and Research Libraries*, pages 280–285, 1971.

[13] William E. McGrath. Predicting book circulation by subject in a university library. *Collection Management*, 1(3–4):7–23, 1976–77.

[14] J.R. Quinlan. Induction of decision trees. *Machine Learning*, 1(1):81–106, 1986.

[15] Stanley J. Slote. Identifying useful core collections: A study of weeding fiction in public libraries. *Library Quarterly*, 41(1):25–34, January 1971.

[16] Stanley J. Slote. *Weeding Library Collections — II*. Libraries Unlimited, Inc., Littleton, CO, 1982.

[17] Richard W. Trueswell. User circulation satisfaction vs. size of holdings at three academic libraries. *College and Research Libraries*, 30(3):204–213, 1971.

[18] William A. Wortman. *Collection Management: Background and Principles.* American Library Association, Chicago, 1989.

A The ID3 Algorithm

The ID3 algorithm is used to choose a dividing criterion for a given node. By applying the ID3 algorithm to the root of a one node decision tree we obtain a decision tree with a root and several children. The ID3 algorithm can be applied recursively to each of the children to create an entire decision tree, terminating when the algorithm determines there is no appropriate dividing criterion for any leaf node of the tree.

Suppose we are considering leaf node N, which has p_N titles. Let q_i be the number of titles in N that were checked out i times in the "recent past." The information inherent in node N, $I(N)$, is defined to be

$$I(N) = -\sum_i \frac{q_i}{p_N} \log_2 \frac{q_i}{p_N}$$

This quantity is measured in *bits*, since it is supposed to represent the number of computer bits needed to store the information in a node.

Suppose we tentatively choose criterion C as a dividing criterion and divide N based on criterion C. Call the children of N $N_1, \ldots, N_v$, where v is the number of values for criterion C. Let p_i be the number of titles N_i inherits from N. The expected information to create the children is defined by

$$E(C) = \sum_{i=1}^{v} \frac{p_i}{p_N} I(N_i)$$

The information gain in dividing node N on criterion C is therefore

$$\text{gain}(N, C) = I(N) - E(C)$$

The use of the term "gain" is perhaps a bit misleading, because while it indeed measures the information gain of the children of N over N itself, it does not take into account the information required to make the division. This statistic can be expressed as

$$IV(N, C) = -\sum_{i=1}^{v} \frac{p_i}{p} \log_2 \frac{p_i}{p}$$

We wish to maximize the quantity

$$\text{gain}(N, C)/IV(N, C)$$

This *gain ratio* statistic has the advantage over the gain statistic in that it does not favor criteria that splinter the data into many criteria, which may make the gain quantity large due merely to the overwhelming magnitude of the summation limit. The gain ratio suffers from its own problem however, in that it may inordinately favor criteria that have a near-zero value of IV. We therefore use a combination of the gain and gain ratio statistics in our final decision algorithm.

Suppose that there are n possible criteria on which to divide node N. We choose the dividing criterion for node N as follows.

- Choose the $n/2$ criteria that have above-average gain for dividing on node N.

Criterion	*Gain*	*Gain ratio*
CHECKOUT HISTORY	.2307 bits	.1231
LAST USE	.1993 bits	.0655
COUNTRY	.1328 bits	.0376
LC CLASS	.1601 bits	.0253
LANGUAGE	.0813 bits	.0246
DATE OF PUBLICATION	.1167 bits	.0167

Table 2: The values calculated for the ID3 decision algorithm when picking a dividing criterion for the root of a decision tree. The algorithm fails only in giving undue weight to COUNTRY.

- Discard those criteria that have 0 gain. If no criteria remain, do not divide node N.

- Otherwise, output that criterion which maximizes the gain ratio $\text{gain}(N, C)/IV(N, C)$.

The gains and gain ratios for each criterion when calculated on the root node of a decision tree are summarized in Table 2. The results predicted by the ID3 algorithm should parallel those of the one-criterion decision trees (Figure 2). For the most part, the ID3 algorithm does well, ranking CHECKOUT HISTORY and LAST USE first, but it inaccurately claims COUNTRY is the next most useful criterion. The algorithm is a heuristic and is not guaranteed to give optimal results.

Notes

[1]This paper is based upon work supported in part by the National Science Foundation under Grant No. IRI-9350192, and in part by an equipment grant from the Harvard University Library. We would like to thank Dale Flecker and Charles Husbands for normalizing and making available the Harvard College Library databases, and Richard De Gennaro, Lawrence Dowler, Dale Flecker, Charles Husbands, Curtis Kendrick, Margo Seltzer, and Michael Smith for their assistance and comments.

[2]Division of Applied Sciences, Harvard University, Cambridge, Massachusetts.

[3]Aggregate statistics of this sort in this paper were provided by Dale Flecker and Curtis Kendrick of the Harvard University Library.

[4]We refer to moving titles off-site, as opposed to individual volumes, because the studies we report on here perform the analysis and choice at that level. The decision was made on the basis of technological expedience: the available bibliographic data are stored on computer by title.

[5]The EAR metric is, for many purposes, too general a metric in that it averages the choice policy's advantage over the entire range of possible depository sizes. For instance, at the Harvard College Library, it is known that the percentage of books stored off-site will be in the range of 15 to 30 percent for many years to come. Thus averaging advantage over the random policy over the full range of 0 to 100 percent may give a misleading estimate of a policy's performance. Where further information of this sort is available, it should be taken advantage of. For the purpose of this paper, we make the weakest possible assumption, that all off-site storage percentages

are equally likely, as the basis for our evaluation.

[6]We use PUBLICATION DATE as an approximate replacement for acquisition date.

[7]Another small change is required in our evaluation procedure. Recall that only leaf nodes are considered when taking titles to put in the depository. With backing off, however, some leaf nodes effectively move their titles to their parent, so some parents of leaf nodes may include titles that need to be considered for the depository. It is not difficult to modify our evaluation algorithm to include the appropriate non-leaf nodes.

[8]Unfortunately, for those titles that are in the system, computerized circulation records for a particular 11-month period were lost. The tests reported here presumably suffer from some inaccuracy due to this missing data as well.

List of Tables

1 Criteria considered in models of predicting book use. The number of values evidenced in an 80,000 title sample is provided in brackets in the second column. 10

2 The values calculated for the ID3 decision algorithm when picking a dividing criterion for the root of a decision tree. The algorithm fails only in giving undue weight to COUNTRY. 43

List of Figures

1 Performance of random and clairvoyant choice policies. These two provide expected extremes in terms of performance. 4

2 Performance of single criterion choice policies. The two past use criteria perform the best overall. 11

3 Detail of Figure 2 for small off-site percentages. In this range LANGUAGE and LC CLASS outperform the past use criteria. The clairvoyant line cannot be seen because it lies along the horizontal axis. . 12

4 Performance of choice policies recommended by Fussler and Simon [4]. Policies using use records outperform those that do not in general, but not when less than 20 percent of books are in the depository. Adding LC CLASS does not always improve performance, perhaps because of overtraining. 14

5 A simple decision tree. 14

6 A single-criterion decision tree that divides on the LAST USE criterion.. 18

7 One of Fussler and Simon's choice policies shown as a decision tree. . 19

8 Performance of LAST USE criterion with subsamples of various sizes. As sample size increases, performance improves slightly as well, quickly converging to a standard value. 20

9 Performance of LAST USE criterion with subsamples of various sizes, detail. 21

10 Performance of a maximal decision tree compared with choice policies recommended by Fussler and Simon [4]. The maximal tree does not perform much better than the others despite its greater complexity. . 25

11 Performance of ID3-ordered tree and a maximal tree. The ID3-ordered tree outperforms the maximal tree with its dividing order picked at random. In addition, it has a smaller variance, making it easier to analyze its efficacy. 27

12 Performance of an ID3-ordered decision tree when tested on the training set and when tested on a separate testing set. The far superior performance of the tree in the former case is evidence of overtraining. 29

13 Effect of smoothing on performance of ID3-ordered trees. The right type of smoothing can dramatically improve performance. 32

14 Performance of a backed-off, ID3-ordered tree compared to trees recommended by Fussler and Simon [4]. The smoothed, ID3-ordered tree is clearly superior to the simpler trees. 33