Performance Metrics and Data Mining for Assessing Schedule Qualities in Paratransit

Romy Shioda, Marcus Shea, and Liping Fu

The simple productivity measures and hard constraints used in many paratransit vehicle scheduling software programs do not fully capture the interest of all the stakeholders in a typical paratransit organization (e.g., passengers, drivers, municipal government). As a result, many paratransit agencies still retain a human scheduler to look through all of the schedules to manually pick out impractical, unacceptable runs. (A run is considered one vehicle's schedule for one day.) The goal of this research was to develop a systematic tool that can compute all the relevant performance metrics of a run, predict its overall quality, and identify bad runs automatically. This paper presents a methodology that includes a number of performance metrics reflecting the key interests of the stakeholders (e.g., number of passengers per vehicle per hour, deadheading time, passenger wait time, passenger ride time, and degree of zigzagging) and a data-mining tool to fit the metrics to the ratings provided by experienced schedulers. The encouraging preliminary results suggest that the proposed methodology can be easily extended to and implemented in other paratransit organizations to improve efficiency by effectively detecting poor schedules.

Demand-responsive transit or paratransit operations involve a number of managerial functions, such as trip reservation, vehicle monitoring, scheduling and dispatching, and business reporting. Among these functions, the most challenging is the scheduling process, which generates vehicle operating schedules by assigning trip requests to a fleet of vehicles. The challenge is mainly caused by the involvement of multiple stakeholders in a paratransit system, including the service provider, the customers, and the drivers, all of whom usually have diverse interests and thus different views of what constitutes a good or bad run. (A run is considered a single vehicle's schedule for a single day.)

From a customer's point of view, a good run is one that would lead to on-time pickup and delivery, minimum diversion from a direct ride, and consistent and familiar routes and drivers. These concerns are typically translated into scheduling constraints such as guaranteed service, maximum ride time, and maximum wait time.

The drivers' perspective on schedules is mostly related to the characteristics of their assigned shifts, such as shift type, availability of appropriate breaks (e.g., for lunch), types of clients assigned (e.g., ambulatory versus wheelchair passengers), the number and types of trips (e.g., long versus short), and, for contract-based service, total income. Some of these concerns can be accommodated, either fully or partially, in the scheduling process (such as breaks and number of trips), whereas others are difficult to explicitly account for (*1*).

Lastly, the service providers and agencies must ensure the financial vitality of their service systems under constrained resources and budget. Maximizing productivity while meeting demand is therefore one of their top operating goals in preparing service schedules. This goal is usually translated into the scheduling objective of minimizing total service hours and vehicle distance.

To account for all these concerns, the scheduling process must take into account multiple system objectives and a large set of system constraints. Complicating the matter further is the fact that many of these objectives conflict with one another; satisfying some would mean sacrificing others. For example, a highly productive schedule with efficient ridesharing and utilization of available vehicle capacity could result in significant delays to passengers. In contrast, a system that emphasized schedule adherence or on-time performance would generally have to sacrifice some productivity to achieve this goal.

A range of different scheduling methods, from manual to fully automated, are being used today in the paratransit industry (*1, 2*). Manual scheduling by human schedulers is still quite popular in paratransit, especially in small- and medium-sized systems. Human schedulers can utilize their experience and knowledge to identify trip patterns and devise near-optimal schedules. They also have the unique ability to deal with qualitative and conflicting goals and find balanced solutions that are acceptable to all stakeholders. Manual scheduling is, however, limited in its ability to handle high-demand service systems (e.g., those with more than 1,000 trips per day). For these systems, a computerized scheduling system that can efficiently solve large, real-world scheduling problems is required. The speed advantages of these systems also mean that more options can be assessed and what-if analyses easily conducted.

Central to most computerized scheduling systems is a set of optimization algorithms that solve a mathematically formulated problem called dial-a-ride problem (DARP). The common formulation of DARP includes a generalized cost function to be minimized and a set of service quality constraints (*3–7*). Because of the fundamental nature of these algorithms, the computer-generated runs usually have a large variation in performance; that is, they include some good runs, some mediocre ones, and a few bad ones. It is often these bad runs that generate customer and driver complaints and create a negative view of the underlying scheduling system.

The performance of computer-aided scheduling systems was investigated by Pagano et al., who found that many operators do not use the scheduling features of the software packages that they purchased and that some "pre- and post-implementation comparisons do not show the kind of dramatic efficiency changes operators have hoped for" (*8*). A later study by Pagano et al. found that computer-aided scheduling and dispatching had a significant influence on the quality

R. Shioda and M. Shea, Department of Combinatorics and Optimization, and L. Fu, Department of Civil Engineering, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada. Corresponding author: R. Shioda, rshioda@uwaterloo.ca.

Transportation Research Record: Journal of the Transportation Research Board, No. 2072, Transportation Research Board of the National Academies, Washington, D.C., 2008, pp. 139–147.
DOI: 10.3141/2072-15

of service of paratransit systems (*9*). A recent study by Fu et al. also studied the impact of scheduling method on the technical efficiency of a paratransit system, finding mixed results (*10*).

To identify problematic, low-quality runs, some agencies and service providers still use human schedulers to check all runs generated by a computer system and screen those that are not acceptable (*2*). The bad runs are then modified by either removing some trips or exchanging trips with other runs. This process is time consuming and tedious and places a significant burden on the human schedulers.

This research attempts to develop a methodology that can be used to evaluate computer-generated runs and identify those that are likely unacceptable from the view of an experienced scheduler. The focus is on testing the feasibility of such a methodology instead of developing a generic tool that can be applied to all systems. Several performance metrics are first introduced that are considered to be reflective of the overall quality of a run with respect to the key interests of the stakeholders. A data mining methodology to fit a run's metrics to its quality as rated by an experienced scheduler is then described. The proposed methodology was implemented and assessed using field data from the Disabled and Aged Regional Transportation Systems (DARTS) of Hamilton, Ontario, Canada.

BACKGROUND AND PROPOSED METHODOLOGY

DARTS is a door-to-door transportation service for the elderly and disabled in the city of Hamilton. DARTS has 135 employees and a fleet of 66 buses and 20 vans directed on variable bus routes by a schedule and dispatch service that serves approximately 8,000 registered passengers. Currently, DARTS uses a computerized scheduling system to generate schedules for daily operations. A human scheduler is, however, employed to manually review all the runs to identify the impractical, unacceptable runs. The goal of this research is motivated by the need to develop a tool that can help the dispatcher rank the computer-generated runs to identify those considered unacceptable.

The proposed methodology is in two parts:

1. Performance metrics and quality rating: All the performance metrics that affect the quality of a run are defined. These are measures that the decision maker would consciously or unconsciously use to rate a run. The schedule is essentially being represented with these quantifiable measures. Also, an employee of DARTS acted as the decision maker by rating past runs on a scale of 1 to 10 (with 1 being the worst and 10 being the best).
2. Data-mining model: Given the performance metrics and the corresponding ratings for a group of schedules, a data-mining tool was fit to model the relationship between the metrics and the ratings. After this tool is refined, it will be used to determine the quality of unrated runs. This final step will allow DARTS to automate its schedule review process.

PERFORMANCE METRICS AND QUALITY RATING

From several discussions with the DARTS team, including the manager, schedulers, dispatchers, and drivers, 28 performance metrics were established. Figure 1 illustrates an example of a run with some of its corresponding performance metrics. These metrics were then grouped into those believed to have a positive effect on the quality of a run, and those believed to have a negative effect on quality of a run. Several examples follow.

Metrics with Positive Effects on Quality

- PSNGR/HR: number of passengers per hour (i.e., the average number of passengers the vehicle services per hour on the given run);
- TRIPS/HR: number of trips per hour (i.e., the average number of stops the vehicle services per hour on the given run). If multiple passengers get picked up or dropped off at a single stop, the stop is counted only once; and
- WC/HR: wheelchair passengers serviced per hour (i.e., the average number of wheelchair passengers the vehicle services per hour on the given run).

Metrics with Negative Effects on Quality

- DEADHEAD: total deadheading time (measures the total time the vehicle runs without a passenger);
- DELAY: total cumulative delay time for passengers (i.e., the difference between the estimated time of arrival and the negotiated arrival time, exceeding 15 min. For example, if the negotiated time is 4:00 p.m. and the estimated arrival time is 4:10 p.m., then there is no delay. However, if the arrival time is 4:20 p.m., then there is a delay of 5 min. This is a cumulative measure over all passengers on the run);
- RIDE TIME: total cumulative ride time for all passengers, which sums up the total duration that each passenger spends on the vehicle;
- RIDE > 45: total cumulative ride time for all passengers exceeding 45 min, which sums up the total duration exceeding 45 min that each passenger spends on the vehicle;
- AVG RIDE: average ride time (i.e., the total duration each passenger spends on the vehicle);
- AVG DIST: average trip distance (i.e., the average distance between two consecutive stops in the run);
- MAX DIST: maximum trip distance (i.e., the maximum distance between any two consecutive stops in the run);
- DISTS ≥ 5k: proportion of distances greater than or equal to 5 km (i.e., the fraction of distances between any two consecutive stops that exceeds 5 km); and
- ZZ(θ, d): zigzag metric [i.e., the proportion of turns that have an angle less than or equal to θ and a distance greater than or equal to d (in meters). It is meant to detect sharp turns or capture runs that go from one part of the city to the far opposite side of the city. For example, if the vehicle goes from point A to point B to point C, this turn would be counted if the angle (AB, BC) ≤ θ and distance (BC) ≥ d. Specific zigzag metrics calculated include those with θ value ranging from 20 to 90 and d value ranging from 0 to 5,000 m].

Table 1 shows the correlation matrix of a selection of these performance metrics. Correlations with 0.4 or higher are highlighted in light gray, and correlations with –0.4 or lower are highlighted in dark gray. Most of the correlations between the performance metrics and ratings seem to fit intuition. For example, PSNGR/HR and TRIPS/HR have positive correlations with the ratings, whereas the AVG DIST, DISTS ≥ 5k, DEADHEAD, and ZZ metrics have significant negative correlations with the ratings. The AVG DIST metric also seems to be highly correlated with the ZZ metrics. Perhaps surprisingly, PSNGR/HR, the main productivity measure, has a relatively weak correlation with the ratings.

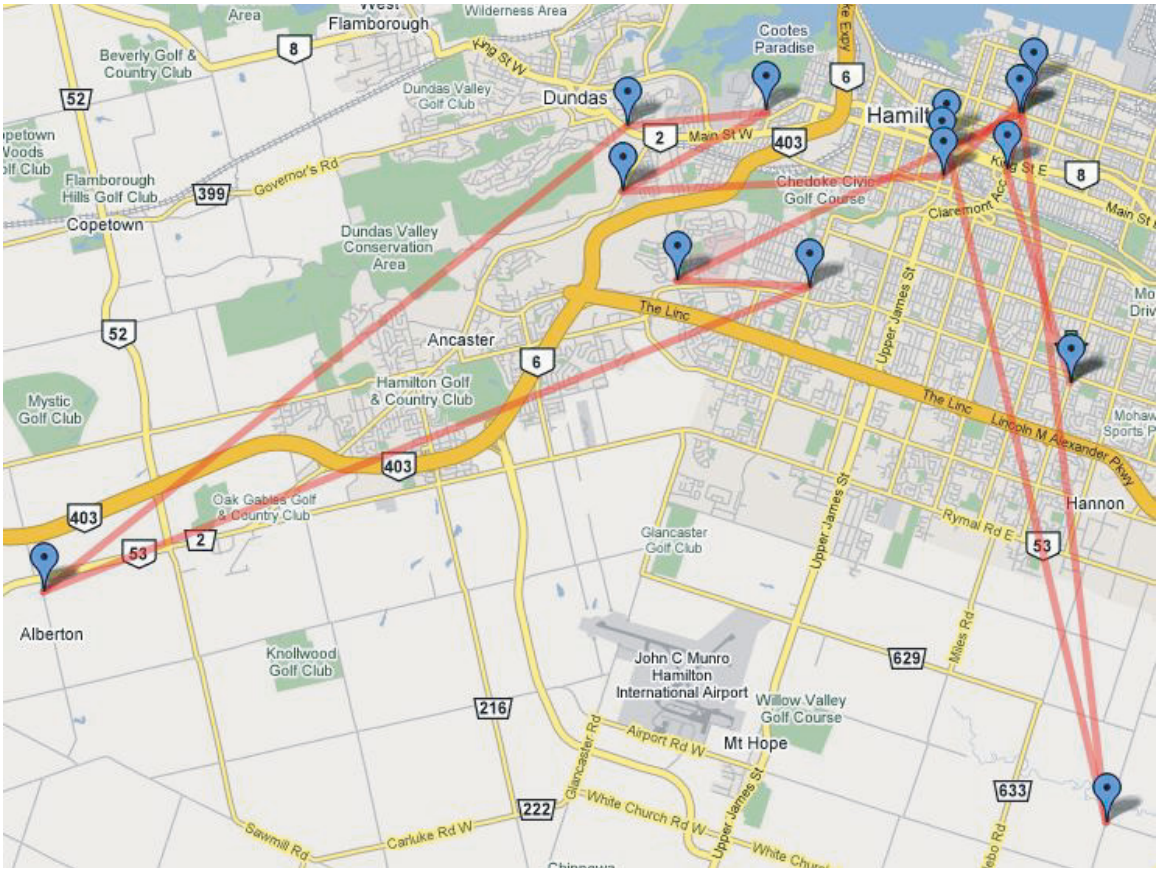

FIGURE 1 A run with PSNGR/HR = 1.049, DEADHEAD = 1.465, DELAY = 0.46, AVG DIST = 6240.78, ZZ(45,2500) = 0.5000, and ZZ(75,5000) = 0.3333.

DATA MINING

This preliminary study used 6 days of data from DARTS. The data consist of 50, 50, 50, 51, 60, and 50 runs per day, with a total of 311 runs. Included are the longitude and latitude of every pickup and dropoff, as well as the negotiated time with the customer and the vehicle's expected arrival time at that stop. Data from 5 of the days are used as the training set to build the data mining models and test the prediction accuracy of these models on the remaining 1-day "hold-out" data. The training set and testing set comprise 261 runs and 50 runs, respectively.

Several regression and classification tools were tested to rank the quality of the runs from worst to best. The performance metrics described above were used as the independent variables in all the models. In the regression models, the ratings are used as the dependent variable. For classification, the ratings are converted to bad or good, where a rating less than or equal to 5 is considered bad and a rating greater than 5 is considered good. This study uses linear regression and regression trees for regression and logistic regression for classification.

The models rank the runs according to their predicted ratings for linear regression and regression trees and according to the predicted likelihood of being a good run for logistic regression. For these ranking algorithms, a cutoff level also needs to be defined to consider all runs with predicted value worse than the cutoff level to be bad. In the implementation phase, the human scheduler ideally needs to examine only those runs that have a predicted value less than the cutoff. This cutoff level should be high enough to capture the majority of the bad runs in the schedule. However, if the cutoff is too high, then the number of bad runs will be overestimated, resulting in less efficient use of the scheduler's time.

To measure the performance of the data-mining models, the false negative rate (FNR) and detection rate (DR) of the models on a variety of cutoff levels are calculated. The FNR is the percentage of truly bad runs among the total number of runs that are predicted good (i.e., with predicted value greater than the cutoff). The DR is the percentage of runs that are predicted as bad. Thus, an effective model would have a cutoff level with a low FNR and a low DR.

Before the results of these models are illustrated, the data preprocessing steps are described.

Data Preprocessing

The data-mining procedure required two data preprocessing steps: data balancing and subset selection.

Data Balancing

Figure 2 illustrates the distribution of the ratings in the training data. Runs with ratings less than or equal to 3 made up less than 3% of the

TABLE 1 Correlation Coefficient Matrix of Performance Metrics

	Rating	PSNGR/ HR	TRIPS/ HR	DEAD HEAD	DELAY	WC/ HR	RIDE TIME	RIDE >45	AVG RIDE	AVG DIST	MAX DIST	DISTS ≥5k	ZZ (45, 1,000)	ZZ (45, 2,500)	ZZ (45, 5,000)	ZZ (75, 1,000)	ZZ (75, 2,500)
PSNGR/HR	0.11	1.00															
TRIPS/HR	0.16	0.69	1.00														
DEADHEAD	–0.33	–0.19	–0.30	1.00													
DELAY	–0.18	0.37	0.16	0.11	1.00												
WC/HR	0.18	0.22	0.15	–0.17	–0.03	1.00											
RIDE TIME	0.10	0.52	0.38	–0.02	0.17	0.21	1.00										
RIDE >45	0.04	0.20	0.17	–0.13	0.09	0.04	0.65	1.00									
AVG RIDE	0.09	–0.10	0.02	–0.19	–0.13	–0.04	0.56	0.66	1.00								
AVG DIST	–0.23	–0.52	–0.65	0.47	–0.03	–0.31	–0.32	–0.04	0.14	1.00							
MAX DIST	–0.04	–0.20	–0.34	0.42	0.05	–0.13	–0.04	0.08	0.10	0.67	1.00						
DISTS≥5k	–0.24	–0.49	–0.57	0.39	–0.03	–0.32	–0.32	–0.08	0.09	0.85	0.41	1.00					
ZZ(45, 1,000)	–0.17	–0.25	–0.33	0.16	0.07	–0.21	–0.26	–0.18	–0.12	0.35	0.12	0.35	1.00				
ZZ(45, 2,500)	–0.22	–0.32	–0.40	0.24	0.05	–0.23	–0.31	–0.17	–0.09	0.46	0.13	0.46	0.86	1.00			
ZZ(45, 5,000)	–0.21	–0.35	–0.43	0.32	0.04	–0.26	–0.32	–0.18	–0.06	0.58	0.24	0.70	0.69	0.78	1.00		
ZZ(75, 1,000)	–0.17	–0.28	–0.39	0.20	–0.01	–0.23	–0.30	–0.15	–0.16	0.42	0.18	0.36	0.72	0.62	0.48	1.00	
ZZ(75, 2,500)	–0.21	–0.40	–0.49	0.28	–0.01	–0.25	–0.38	–0.16	–0.11	0.56	0.22	0.50	0.63	0.79	0.59	0.81	1.00
ZZ(75, 5,000)	–0.26	–0.43	–0.53	0.38	0.00	–0.28	–0.35	–0.15	–0.04	0.69	0.32	0.80	0.51	0.63	0.84	0.60	0.73

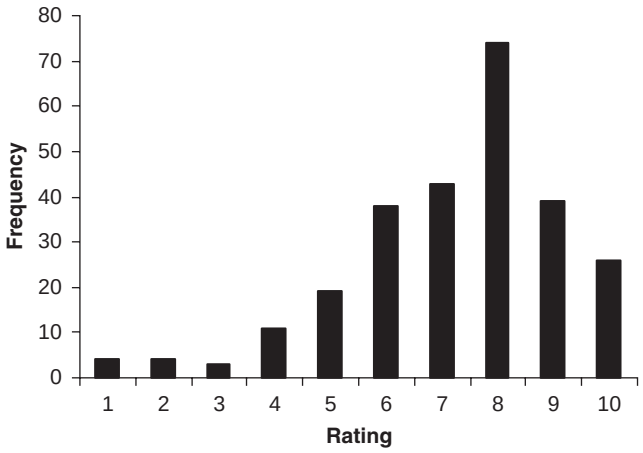

FIGURE 2 Distribution of the ratings in training data.

training data, yet these are the very runs that need to be detected. A trivial prediction model can predict all runs as being good (i.e., with ratings of 6 or higher). Such a model would have high prediction accuracy, but would be unable to differentiate between good and bad runs. To bring more emphasis to the lower-rated runs, the models were trained on two additional data sets using different sampling methods commonly used in data mining and machine learning (*11–13*). In the undersampled set, four runs were randomly sampled from each rating category. Thus, each rating category has four runs, except rating 3, for which only three runs were supplied. This training set allows for equal distribution of the runs and has the advantage of not changing the odds ratio in logistic regression (*14*). The clear disadvantage is that very few runs are left—the training set now has only 39 runs. In the oversampled or in the undersampled set, 26 runs are sampled from each rating category. This technique is oversampling the underrepresented rating category and undersampling the overrepresented categories. This data set contains 260 runs, which is roughly equal to the original size. The testing set is unchanged, with 50 data points—eight of them being bad runs (i.e., with ratings less than or equal to 5).

Subset Selection

There are 28 performance metrics calculated for each run; however, only a fraction of these may be critical in predicting the quality of a run. Primarily for robustness purposes—that is, to limit the variance of the predicted dependent variable—it is desirable to use a small subset of the independent variables (*15, 16*). Thus, besides running the data-mining tools on the full model, a subset of the variables is also selected. The following are the different variable subsets tested:

1. Full model: includes all 28 performance metrics.
2. SubsetA: includes only six metrics: PSNGR/HR, DEADHEAD, DELAY, AVG DIST, ZZ(45, 2,500), ZZ(75, 5,000). These metrics were chosen after discussions with DARTS and after initial empirical testing.
3. SubsetB: includes only three metrics: DEADHEAD, DELAY, ZZ(45, 2,500). These metrics were chosen after discussions with DARTS and after initial empirical testing.

For the linear and logistic regression models, both forward and backward selection were also tested. However, using the full model, either SubsetA or SubsetB, or both, had superior prediction performances to forward and backward selection in all cases. Thus, the results of the forward and backward selection are not presented in this work.

Linear Regression

SPSS software is used for the linear least squares regression model. Out of all the different models tested, the model using the oversampled or the undersampled data set with SubsetA of the performance metrics appears to perform best (*17*). The root mean square and mean absolute error of the training set are 2.2136 and 1.8132, respectively, with an R^2 of 0.406. The root mean square and mean absolute error of the test set are 2.3137 and 1.8680, respectively. Figure 3 illustrates the FNR and the DR of the model with varying cutoff levels on the testing set. To get a 0% FNR, a minimum cutoff level of 6.299 is needed, which has a corresponding DR of 46.81%; in other words, to capture all the bad runs in the testing set, 46.81% of the lowest rated testing set runs would need to be examined. Table 2 illustrates the corresponding linear regression model.

Regression Trees

A regression tree is a decision tree approach on regression in which a binary tree is built so that the data are split according to a split decision (e.g., DEADHEAD ≤ 20 or not, PSNG/HR ≤ 1.0 or not) at each nonleaf node (*18, 19*). At a leaf or terminal node, it builds a linear regression model. The aim is to build such a tree that the actual dependent variable value and predicted dependent variable value of each training set data are as close as possible, often in terms of squared errors. GUIDE is used as the regression tree software (*20*).

The regression tree method worked best on the undersampled data set with a full set of variables. The root mean square and the mean absolute error of the training set are 2.3901 and 1.9992, respectively, with an R^2 of 0.311. The root mean square and the mean absolute error of the testing set are 2.5717 and 2.1944, respectively. Figure 4 illustrates the FNR and the DR on the testing set with various cutoff levels. To get a 0% FNR, a minimum cutoff level of 5.26 is needed, which corresponded to a 36% DR. After pruning of the tree using 10-fold crossvalidation, the resulting decision tree has only one node—that is, it ultimately simplifies to a linear regression model. The resulting regression model used only two variables: DELAY and ZZ(45, 1,000). The linear regression of the regression tree model (undersampling with SubsetA) is shown below:

	Coefficient	*t-Statistic*	*p-Value*
(Constant)	5.483	13.747	0.000
DELAY	–0.883	–2.057	0.047
ZZ(45, 1,000)	–1.520	–3.789	0.001

Logistic Regression

SPSS software is used for the binary logistic regression model. Out of all the different models tested, the model using the oversampled or the undersampled data set with SubsetA of the variables performed best (*17*). The model has a –2 log likelihood of 281.125, a Cox and Snell R^2 value of 0.251, a Nagelkerke R^2 value of 0.335, and a chi-square value of 75.311 with significance under 0.0001. Figure 5 illustrates the FNR and DR on the testing set. All the bad runs were captured

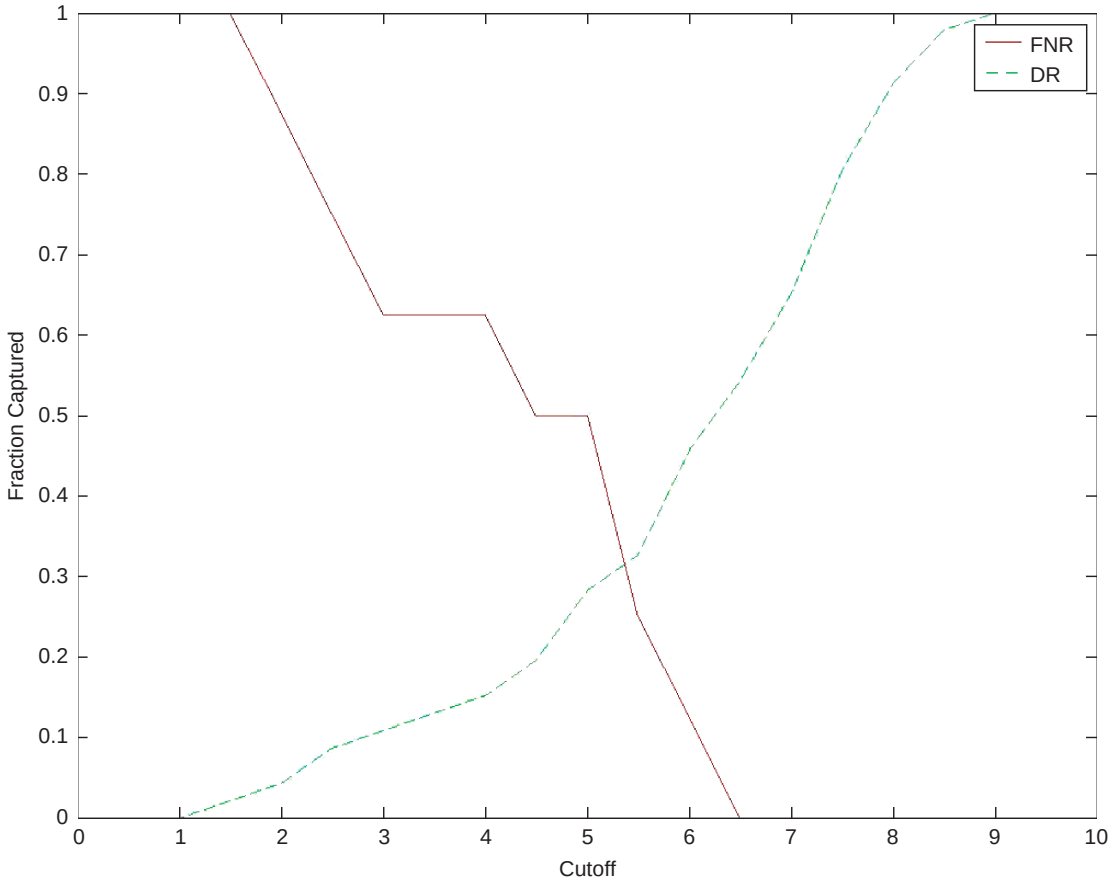

FIGURE 3 FNR and DR of linear regression model.

with a cutoff level of 0.562, which corresponded to a 52% DR. Details of the model are shown in Table 3.

Discussion

It is difficult to say whether one data-mining model dominates over the others. It appears that all three models are able to rank the testing set runs so that the top 50% of the lowest-ranked runs capture all of the actual bad runs. The regression results from the regression tree appear to be most efficient, in that it is able to capture all of the bad runs with a 36% DR. This implies that a human scheduler using this model on the testing data would have seen all eight bad runs in the testing set by examining only the worst 36% of the runs. Because the model simplifies to a linear regression model and is easy to implement, the resulting model is implemented in this methodology.

TABLE 2 Linear Regression Model: Oversampling and Undersampling with SubsetA

	Coefficients	*t*-Statistic	*p*-Value
(Constant)	5.507	39.539	0.000
DEADHEAD	–0.300	–1.868	0.063
Waittime15	–0.813	–5.593	0.000
ZZ(45, 2,500)	–0.868	–4.508	0.000
PSNGR/HR	–0.045	–0.269	0.788
AVG DIST	0.103	0.488	0.625
ZZ(75, 5,000)	–1.019	–4.139	0.000

IMPLEMENTATION

A Java-based implementation of the methodology was developed that reads in a set of run data and then assigns ratings for these runs on a scale of 0 to 100. These ratings are simply scaled from the 1 to 10 rating outputted from the earlier linear regression, solely for ease of use. The runs can then be sorted either chronologically or by rating, and the program can output a Microsoft Excel report. If the runs are sorted by ratings, the user can determine the appropriate cutoff level.

Along with the rating, the program will also output a description for each run. Descriptions will consist only of negative comments on the run that are generated by looking at each metric for the run. If a metric is more than 1.5 standard deviations away from the mean in the nonfavorable direction, then it outputs a description indicating that this metric may be partly responsible for the poor rating. This tolerance of 1.5 can easily be modified by the user. This feature was received very favorably by the manager of DARTS. See Figure 6 for the graphical user interface.

The methodology also includes a feedback process that allows DARTS dispatchers to update the current data, helping to ensure that future runs will be rated more correctly. If dispatchers are unsatisfied with a predicted rating, they can press the Edit Rating button to update

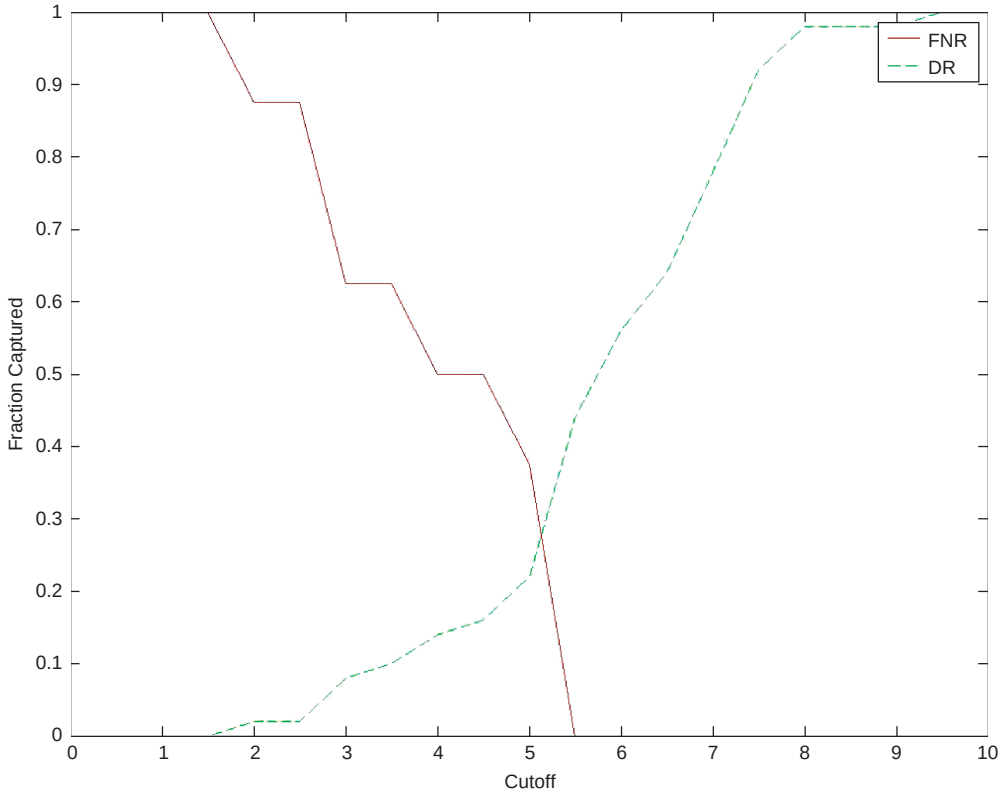

FIGURE 4 FNR and DR of regression tree model.

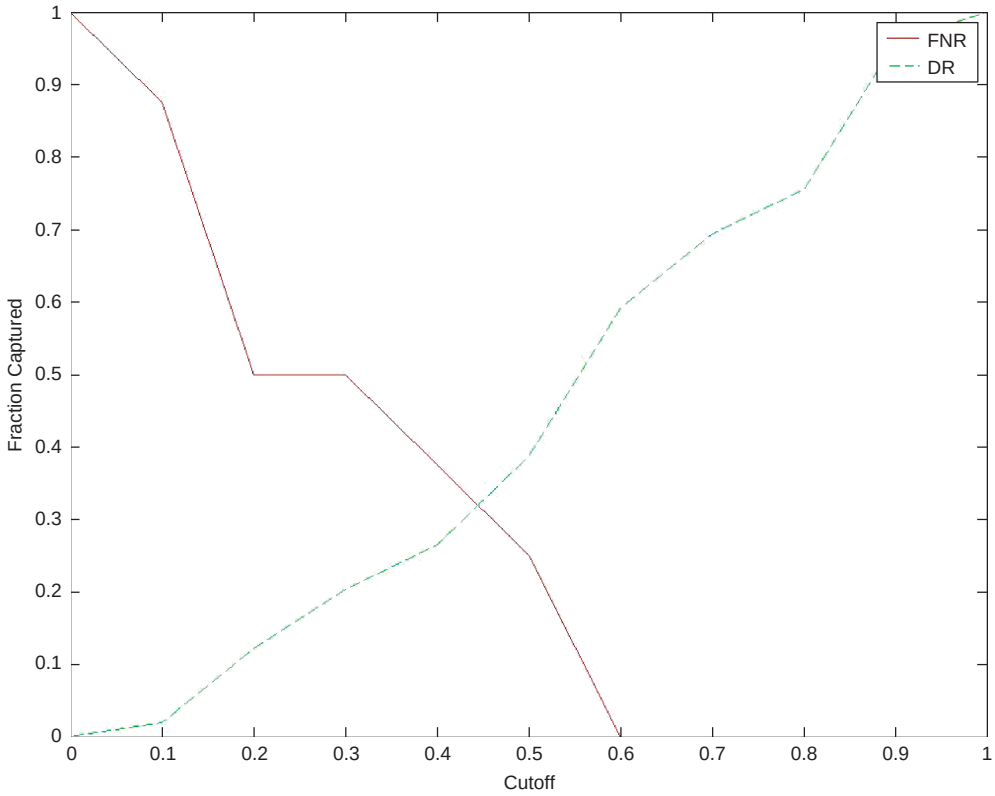

FIGURE 5 FNR and DR of logistic regression model.

TABLE 3 Logistic Regression Model: Oversampling and Undersampling with SubsetA

	Coefficients
(Constant)	–0.362
DEADHEAD	–0.373
Waittime15	–0.580
ZZ(45, 2,500)	–0.235
PSNGR/HR	0.008
AVG DIST	–0.683
ZZ(75, 5,000)	–0.010

that run's rating. The dispatcher can also choose to not rate a run at all by pressing Remove Run. When the dispatchers are satisfied with all the ratings, they can press Add to Training Data to add the currently rated runs into the training data.

CONCLUSION

The goal of this study was to build an automated system to detect bad runs. Such a tool can aid paratransit organizations in cutting down the time needed by human schedulers to filter out these bad runs. Key aspects of this approach are defining and calculating appropriate quantitative performance metrics, having a decision maker at the organization rate runs, and building a prediction model that can link the performance metrics to the corresponding ratings.

This preliminary work illustrates an example of such a technique with many direct extensions. For example, only a few data-mining models have been explored—in the future, the data could be tested on more sophisticated data-mining and machine learning tools. In addition, the data preprocessing step could be extended to remove outliers in the data and consider additional variable subsets. There appeared to be several errors in the ratings of the data set, yet because of the limited amount of data, they could not be omitted. More data along with a method to remove potentially erroneous data will likely produce significantly better prediction accuracy from all models.

In addition, different paratransit organizations have different characteristics that may require additional or modified performance metrics. Thus, the user needs to work with the decision makers to ensure that all essential metrics are captured in the model.

This approach is a flexible and simple framework for detecting poor runs. Even with the very small data set, encouraging prediction results could be seen. Thus, it seems that such a methodology can be easily extended and implemented in many other paratransit organizations around the world.

ACKNOWLEDGMENT

The authors thank the Disabled and Aged Regional Transportation System of Hamilton, Ontario, for providing them with guidance and data.

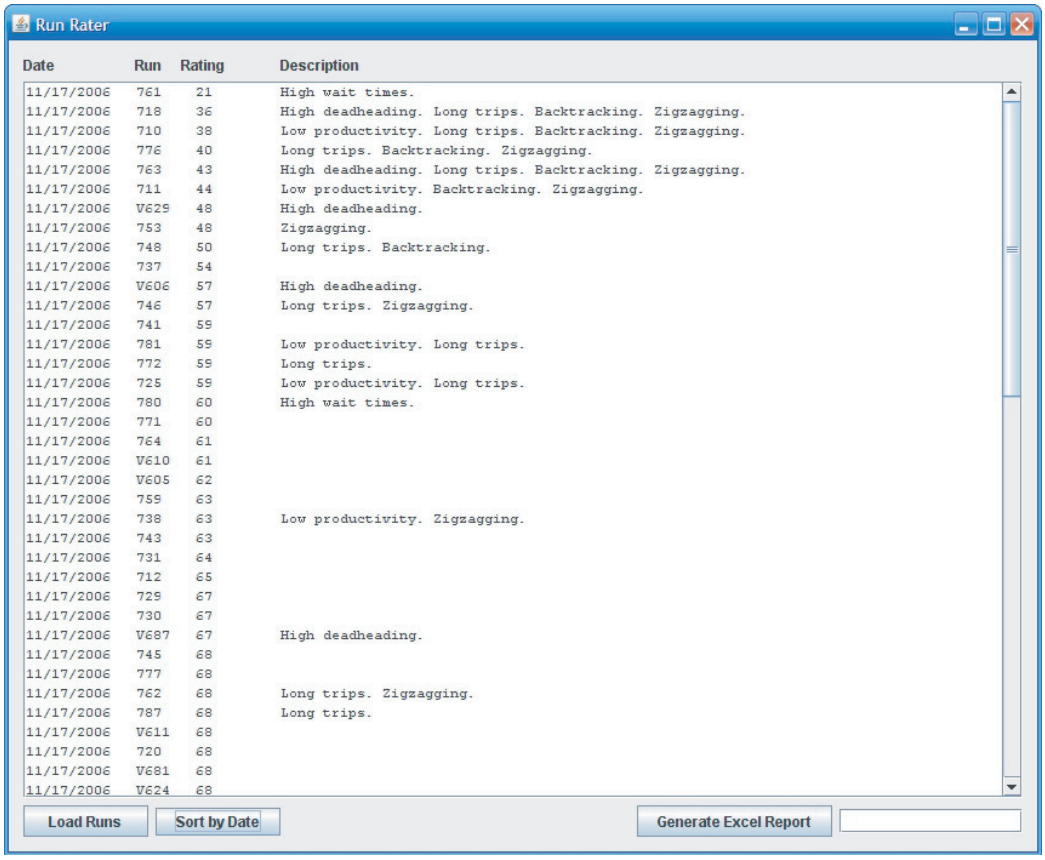

FIGURE 6 User interface of the implementation.

REFERENCES

1. Lave, R. E., R. Teal, and P. Piras. *TCRP Report 18: A Handbook for Acquiring Demand-Responsive Transit Software.* TRB, National Research Council, Washington, D.C., 1996.
2. Kessler, D. S. *TCRP Synthesis of Transit Practices 57: Computer-Aided Scheduling and Dispatch in Demand-Responsive Transit Services.* TRB, National Research Council, Washington, D.C., 2004.
3. Wilson, N. H. M., and N. H. Colvin. *Computer Control of the Rochester Dial-a-Ride System.* Report R77-31. Department of Civil Engineering, Massachusetts Institute of Technology, Cambridge, 1977.
4. Bodin, L., B. Golden, A. Assad, and M. Ball. Routing and Scheduling of Vehicles and Crews: The State of Art. *Computers and Operations Research,* Vol. 10, 1983, pp. 69–211.
5. Jaw, J., A. Odoni, H. N. Psaraftis, and N. H. M. Wilson. A Heuristic Algorithm for the Multi-Vehicle Advance Request Dial-a-Ride Problem with Time Windows. *Transportation Research,* Vol. 20, No. 3, 1986, pp. 243–257.
6. Fu, L. Scheduling Dial-a-Ride Paratransit Under Time-Varying Stochastic Congestion. *Transportation Research,* Vol. 36, No. 6, 2002, pp. 485–506.
7. Cordeau, J. F., and G. Laporte. The Dial-a-Ride Problem: Models and Algorithms. *Annals of Operations Research,* Vol. 153, 2007, pp. 29–46.
8. Pagano, A. M., P. Metaxatos, and M. King. How Effective Is Computer-Assisted Scheduling and Dispatching in Paratransit? Results from a Survey. In *Transportation Research Record: Journal of the Transportation Research Board, No. 1760,* TRB, National Research Council, Washington, D.C., 2001, pp. 100–106.
9. Pagano, A. M., P. Metaxatos, and M. King. Impact of Computer-Assisted Scheduling and Dispatching Systems on Paratransit Service Quality. In *Transportation Research Record: Journal of the Transportation Research Board, No. 1791,* Transportation Research Board of the National Academies, Washington, D.C., 2002, pp. 51–58.
10. Fu, L., J. Yong, and J. Casello. Quantifying the Technical Efficiency of Paratransit Systems Using the Data Envelopment Analysis Method. In *Transportation Research Record: Journal of the Transportation Research Board, No. 2034,* Transportation Research Board of the National Academies, Washington, D.C., 2007, pp. 115–122.
11. Chawla, N. V., K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. SMOTE: Synthetic Minority Over-Sampling Technique. *Journal of Artificial Intelligence Research,* Vol. 16, 2002, pp. 321–357.
12. Kubat, M., and S. Matwin. Addressing the Curse of Imbalanced Training Sets: One-Sided Selection. *Proc., 14th International Conference on Machine Learning,* Nashville, Tenn., 1997, pp. 179–186.
13. Ling, C., and C. Li. Data Mining for Direct Marketing Problems and Solutions. *Proc., 4th International Conference on Knowledge Discovery and Data Mining (KDD-98),* New York, 1998, pp. 73–79.
14. Hosmer Jr., D., and S. Lemeshow. *Applied Logistic Regression.* John Wiley & Sons, Hoboken, N.J., 2000.
15. Miller, A. *Subset Selection in Regression.* Chapman and Hall/CRC, Boca Raton, Fla., 1990.
16. Ryan, T. P. *Modern Regression Methods.* Wiley Series in Probability and Statistics. John Wiley & Sons, Hoboken, N.J., 1997.
17. SPSS, Inc. home page. www.spss.com.
18. Breiman, L., J., Friedman, R. Olshen, and C. Stone. *Classification and Regression Trees.* Chapman and Hall/CRC, Boca Raton, Fla., 1984.
19. Loh, W.-Y. Regression by Parts: Fitting Visually Interpretable Models with GUIDE. In *Handbook of Computational Statistics,* vol. 3, Springer, Berlin, Germany, 2008, pp. 447–469.
20. GUIDE Regression Tree. www.stat.wisc.edu/~loh/guide.html.

The Paratransit Committee sponsored publication of this paper.