A Benchmark for Predicting Turnaround Time for Trucks at a Container Terminal

Sjoerd van der Spoel – Chintan Amrit – Jos van Hillegersberg

University of Twente
7500 AE Enschede
The Netherlands
{s.j.vanderspoel,c.amrit,j.vanhillegersberg}@utwente.nl

ABSTRACT: *Creating a reliable predictive model is a vital part of business intelligence applications. However, without a proper benchmark, it is very difficult to assess how good a predictive model really is. Furthermore, existing literature does not provide much guidance on how to create such benchmarks. In this paper we address this gap by presenting a method for creating such a such a benchmark. We demonstrate the method by developing predictive models for truck turnaround time, created using both regression and classification methods. We use data generated in a simulated terminal for developing these models. We establish the parameters and parameter distributions of the simulation through a structured review of the relevant literature. We show that congestion, start time and route through the terminal together are good predictors of turnaround time, leading to adequate predictive performance. These results can then be used as a benchmark for predictive models on truck turnaround time, thereby demonstrating our general method for creating such benchmarks.*

KEY WORDS: *Container terminal, benchmark, turnaround time, predictive analytics*

1. Introduction

When developing a predictive model, various metrics are used to evaluate its predictive power (Shmueli, 2010; Shmueli & Koppius, 2011). Predictive model development is an iterative process, where the models are improved until *saturation* is reached, i.e. until the metrics no longer improve with changes to the model. As there is often no theoretical 'best' or even 'good' performance known for the domain, it is difficult to assess how good the final predictive model *is* with regard to how good it *could* be. We resolve the issue of comparing predictive model performance, by developing a benchmark predictive model.

The domain we develop the benchmark for is *predicting truck turnaround time* at a container terminal. Truck turnaround time is the total time a truck spends between entering and exiting a terminal. Transport companies in the Port of Rotterdam often make multiple terminal visits per day. For these companies, accurately predicting turnaround time means that they can plan the order of terminal visits to minimize total turnaround time. Therefore, turnaround time of a truck is an important measure for trucking companies, but we could find no existing models in literature to predict turnaround time. Hence the need for a benchmark predictive model.

Shmueli & Koppius' (2011) method provides guidelines on how to develop predictive models, such as how to select variables and how to do data cleaning. We combine Shmueli & Koppius' (2011) method with a method for generating data on turnaround time that is realistic but not influenced by soft factors (human factors, such as behaviour, politics or trust) . This data is used to create one or more predictive models. We then record the performance (e.g. accuracy or mean error) of these models.

These performance figures serve as a benchmark: as soft factors are likely to cause noise in data, which negatively affects the performance of predictive models (Van der Spoel, Amrit, & Van Hillegersberg, 2015, 2013). The benchmark models, which are not affected by these factors, would therefore represent good results for the domain.

This paper is divided into two parts. First, we present a method for generating data on truck turnaround time in Section 2. This method consists of a combination of a structured literature review on the domain of turnaround time and a simulated container terminal. Second, in Section 3, we use the data from this simulated terminal to develop predictive models. It is the performance results of these models that forms our benchmark for predicting truck turnaround time.

2. Data generation

In this section, we present our method for generating data for a benchmark predictive model. We want the data for the benchmark to reflect the actual situation at a container terminal as best as possible. This means that we need to find what aspects of a terminal affect truck turnaround time, for which we use a structured literature review, presented in the document: `https://goo.gl/r38F2z`. That way, the aspects we use are well grounded in established theory.

2.1. Simulated terminal

The variables related to truck turnaround time were taken from the papers selected through our literature review. These are used to create the simulated terminal and to select variables for the predictive model (see section 3.5).

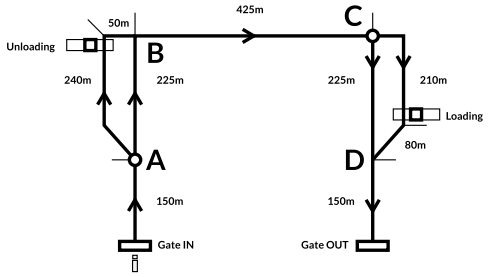

Figure 1. *Layout of a simple container terminal with one unloading and one loading crane*

The variables from the literature review are represented in the simulated terminal as follows: congestion, demand, arriving vessels and time of day are represented by *truck arrival distribution*; capacity and amount of equipment, operator style and error, job duration, gate processing rate, stack configuration, and storage sequence are represented by *crane & gate processing rate*; the number of containers to (un)load and travel distance are represented by the *route through the terminal*. Weather and reefer containers were not implemented in the simulation.

Figure 1 shows the basic structure of our simulation, the layout of a simple terminal. The layout is based on actual terminals in the port of Rotterdam in the Netherlands. Each truck arriving at the terminal has to go through the GATE IN, before any empty containers are removed at UNLOADING by a yard crane (YC) and a new container is put on the truck by another YC at LOADING. The figure also shows the distances between points on the terminal, and the different routes a truck can take depending on whether it will only load, only unload, or both load and unload a container.

3. Predictive model development

3.1. Goal description

The performance of the predictive model we are developing serves as a benchmark. We therefore, record a wide range of metrics, so that the results can be compared to as many other predictive models as possible. For *classification*, the main metric is accuracy: the number of correctly classified instances divided by the total number of instances. As we run multiple tests of each predictive model, we also record the distribution of accuracy across these tests.

For *regression*, the metrics are all error metrics: they measure the difference between the predicted turnaround time and actual turnaround time for each instance. We record the following metrics for regression: *mean error*, *mean absolute error*, *mean square error*, *root mean square error*, and *mean absolute percentage error*.

3.2. Data collection & study design

We use the simulated terminal from Section 2.1 to generate data to create (train) and test a predictive model. The number of datasets we generate is based on two considerations. First, as mentioned in the previous section, there are two variations in the distribution of truck arrivals, meaning that at least two datasets should be tested. Second, as all the selected papers from our review suggest that either congestion or terminal capacity are relevant, we also generate a congested and an uncongested dataset, bringing the total to four.

To construct an uncongested/a congested dataset, we vary the total number of trucks arriving per given time period. The terminal is congested if one or more queues have a higher input flow than output flow. Given that the least-capacity queues (the in- and out gates) can process around 20-25 trucks per hour, this means that the total number of trucks arriving per hour should be greater than 25 at some points in time for the congested dataset, and never greater than 20 for the uncongested dataset.

Each simulation will run for 1000 hours of simulated time (41 days and 16 hours), in order to make sure that the simulation reaches a 'congested' state. During that time, some 15000 to 30000 trucks arrive at the terminal, depending on the truck arrival distribution.

The available variables for each terminal visit in the dataset (as per the findings of the literature review) are the following: *start time*, i.e. the time the truck entered the terminal's first queue (in seconds); *end time*, i.e. the time the truck left the terminal's last queue (in seconds); *load*, whether the truck picked up a container (true or false); *unload*, whether the truck dropped off a container (true or false); and *congestion*, the total number of trucks in a queue divided by the total capacity of queues *at the time the truck entered the terminal* (ranging from zero to one).

3.3. Data preparation

With regard to dataset partitioning, we will use a ten-fold cross validation. In this approach, a dataset is split randomly into ten equal size partitions. There are then ten iterations over the data, each time selecting one partition as holdout and the other nine as training set. After ten iterations, we average teh results, with each partition will have been used as a holdout set exactly once. This approach to partitioning should eliminate bias/"overoptimistic predictive accuracy"(Shmueli, 2010).

3.4. Exploratory data analysis

From the distribution of turnaround time for the arrival distributions, we find that the uncongested β arrival distribution results in turnaround times that fit a β distribution [1], albeit with different parameters. None of the other turnaround distributions fit a well-known distribution, i.e. a normal, exponential, lognormal, logistic, gamma or uniform distribution.

There do not seem to be major differences between the turnaround times per order type (loading, unloading or both). The exception is **??**, where unloading seems have its peak values at shorter turnaround times than loading or unloading & loading.

Congestion describes how much of the total capacity of the queues along a truck's route is filled. For each truck arrival distribution, turnaround time goes up as congestion goes up. This relation is significant at the $p < 0.01$-level, with $0.66 \leq R^2 \leq 0.95$

3.5. Choice of variables

The basis for variable selection are the four datasets discussed in Section 3.2, each differing in the type of arrival distribution and the amount of congestion.

Measurement quality is about the quality of the data used. As stated in the previous section, the data is fully clean, i.e. the generated data is free of noise. The EDA also shows that at least one of the variables in the dataset (congestion) is a good predictor of turnaround time

With regard to *ex ante availability*, there is one variable that might not be available at the time of prediction, and that is *congestion*. As congestion can only be directly measured by a terminal operator, the availability of this variable strongly depends on the willingness of the terminal operator to supply that data. To take this into account in our benchmark, we will use both a dataset that contains all three variables (route, start time, and congestion) and one that only uses start time and route.

3.6. Choice of methods

This section describes the data mining methods used to make a predictive model from the four datasets. As our goal is to create a predictive model that is as good as possible, we will use several different methods, to determine which performs the best for each dataset.

Based on what previous research (Van der Spoel et al., 2015, 2013) found to be a well-performing data mining algorithm, we use Breiman's Random Forests (L. Breiman, 2001) and a variation on Breiman *et al.*'s Classification and Regression Trees (CART)

[1] To fit a β distribution, which only has values between 0 and 1, the probability density function is multiplied by $\max(t_{turnaround})$

(Leo Breiman, Friedman, Stone, & Olshen, 1984)[2]. Finally, for regression tests, we also test linear regression, as the EDA has shown significant correlations between the predictors and turnaround time.

3.7. Evaluation

All predictive models were developed using the *R* suite of statistical software (R Core Team, 2013), and the *R* implementations of Random Forests (Liaw & Wiener, 2002) and CART (Therneau, Atkinson, & Ripley, 2015). In this section, we present the performance results of each predictive model.

Classification models Table 1 shows the performance for each classifier in each of the sixteen classification tests. Each value indicates how many rows of data were correctly classified as a fraction of the whole test set. Correct implies that the algorithm's prediction of the 5-minute interval of a trucks turnaround time matches the actual interval of the truck's turnaround time.

We find that Random Forest is the better performing of the two algorithms, but not by a big difference. The mean performance for the β arrival distribution datasets is around 0.48, and around 0.35 for the High-low-high arrival distribution datasets. Only using the start time and the route through the terminal has a strong effect on accuracy, reducing it by 30 to 50 percent.

Regression models Table 2 shows the performance for each regression method in each of the twenty-four regression tests. Each value represents a different error statistic, expressed in seconds. Compared to the classification results, there is not one method that stands out over the others, although linear regression seems to perform better than the other algorithms. This can be explained by the significant relations between the predictors and turnaround time (see Section 3.4).

Similar to the classification results, the use of only route and start time as predictors *generally* negatively influences performance. Exceptions are the mean error, which is in fact the worst performing of all, when using only these predictors for the β-datasets.

4. Discussion

In this paper, we set out to demonstrate how one can create a benchmark predictive model for a particular case. We then created a benchmark based on several performance metrics for predicting turnaround time. The predictive models show that low congestion makes

[2] An explanation of the workings of these algorithms goes beyond the scope of this paper, and should be considered a textbook matter

for the best predictions, which is in line with literature. The best accuracy for classification was around 46 percent for the uncongested dataset with a β arrival distribution. The best regression models have a mean absolute error of around 195 seconds, again for the uncongested dataset with a β arrival distribution.

In our simulation we avoid factors such as behavior or politics or other human factors (Amrit, Daneva, & Damian, 2014), that can create noise and significantly affect the outcome of a prediction (Van der Spoel et al., 2015, 2013). These potentially noisy factors make it difficult to compare the performance of predictive models, as their presence, importance and impact strongly depends on its domain (i.e. the specific organization or case for which the model was developed) (Checkland, 1989; Forrester, 1994).

The extent to which a domain is affected by factors such as behavior and politics is determined by the complexity of the domain (Checkland, 1989; Forrester, 1994). A complex domain is one that is open to its environment, and that is subject to behavioral influences (Jackson & Keys, 1984).

In short, a real-world instance of the domain used for our benchmark is likely to be a complex domain. This means that it will be affected by noise from factors such as behavior, politics, and the environment of the domain. As our simulated container terminal represents a simple domain, our benchmark predictive models were developed with data that were not affected by any of these factors. Furthermore, it has a minimum set of variables that are likely to be found in any instance of the domain. This means that the results of the benchmark predictive models serve as a good comparison for instances of the container terminal domain.

5. Conclusion

In this paper, we have demonstrated a method to create benchmarks for predictive model development. We have presented benchmark results for predicting turnaround time for a truck at a container terminal. Turnaround time is the total time it takes a truck to unload and/or load a container at the terminal. The key aspect of our benchmark results is that they are based on a noise-free dataset constructed using a simulated terminal. This data source means that the predictions and predictive power are not affected by case-specific 'noise' such as human behavioral influences and politics (Amrit et al., 2014; Van der Spoel et al., 2013). We used a simulated terminal based on literature to create four datasets, each with minor differences in terms of how many trucks arrive per hour, and how much time is in between the individual truck arrivals. We therefore construct a benchmark that is usable for a broad range of specific case situations.

We used the four datasets to train predictive models for the purposes of both regression and classification. The performance of the predictive models was measured with a multitude of metrics, again to make sure that the benchmark serves as a comparison for a wide range of situations.

The benchmarks developed in this paper are especially usable a comparison for complex domains, where behavior, myths, meaning, politics and environmental influences lead to noise. There, the benchmark provides clean results for comparison, that are not affected by any of these factors.

References

Amrit, C., Daneva, M., & Damian, D. (2014). Human factors in software development: on its underlying theories and the value of learning from related disciplines. a guest editorial introduction to the special issue. *Information and Software Technology*, *56*(12), 1537–1542.

Breiman, L. [L.]. (2001). Random forests. *Machine Learning*, *45*(1), 5–32.

Breiman, L. [Leo], Friedman, J., Stone, C. J., & Olshen, R. A. (1984). *Classification and regression trees*. CRC press.

Checkland, P. B. (1989). Soft systems methodology. *Human systems management*, *8*(4), 273–289.

Forrester, J. W. (1994). System dynamics, systems thinking, and soft or. *System Dynamics Review*, *10*(2–3), 245–256.

Jackson, M. & Keys, P. (1984). Towards a system of systems methodologies. *The Journal of the Operations Research Society*, *35*(6), 473–486.

Liaw, A. & Wiener, M. (2002). Classification and regression by randomforest. *R News*, *2*(3), 18–22. Retrieved from http://CRAN.R-project.org/doc/Rnews/

R Core Team. (2013). *R: a language and environment for statistical computing*. Visited on March 13th, 2015. R Foundation for Statistical Computing. Vienna, Austria. Retrieved from http://www.R-project.org/

Shmueli, G. (2010). To explain or to predict? *Statistical Science*, 289–310.

Shmueli, G. & Koppius, O. R. (2011). Predictive analytics in information systems research. *MIS Quarterly*, *35*(3), 553–572.

Therneau, T., Atkinson, B., & Ripley, B. (2015). *Rpart: recursive partitioning and regression trees*. R package version 4.1-10. Retrieved from https://CRAN.R-project.org/package=rpart

Van der Spoel, S., Amrit, C., & Van Hillegersberg, J. (2015). Predictive analytics for truck arrival time estimation: a field study at a European distribution center. *International Journal of Production Research*, *forthcoming*.

Van der Spoel, S., Van Keulen, M., & Amrit, C. (2013). Process prediction in noisy data sets: a case study in a Dutch hospital. In *Data-driven process discovery and analysis* (pp. 60–83). Springer.

Arrival distribution	*Congested*	*Variables*	*Dataset*	*Method*	*Min.*	*1st Qu.*	*Median*	*Mean*	*3rd Qu.*	*Max.*
β	Yes	C,R,S	1	Random Forest	**0.488**	**0.490**	**0.491**	**0.492**	**0.493**	**0.502**
		C,R,S	1	CART	0.452	0.454	0.456	0.456	0.457	0.461
		R,S	2	Random Forest	0.365	0.365	0.365	0.365	0.365	0.365
		R,S	2	CART	0.365	0.365	0.365	0.365	0.365	0.365
	No	C,R,S	3	Random Forest	**0.452**	**0.466**	**0.467**	**0.467**	**0.475**	**0.477**
		C,R,S	3	CART	0.447	0.453	0.454	0.456	0.459	0.468
		R,S	4	Random Forest	0.257	0.257	0.257	0.257	0.257	0.257
		R,S	4	CART	0.257	0.257	0.257	0.257	0.257	0.257
High-low-high	Yes	C,R,S	5	Random Forest	**0.350**	**0.356**	**0.363**	**0.364**	**0.371**	**0.382**
		C,R,S	5	CART	0.247	0.255	0.256	0.256	0.258	0.264
		R,S	6	Random Forest	0.194	0.198	0.200	0.200	0.202	0.203
		R,S	6	CART	0.193	0.193	0.193	0.193	0.193	0.193
	No	C,R,S	7	Random Forest	**0.328**	**0.340**	**0.341**	**0.342**	**0.350**	**0.354**
		C,R,S	7	CART	0.237	0.253	0.254	0.254	0.260	0.263
		R,S	8	Random Forest	0.110	0.112	0.114	0.114	0.115	0.117
		R,S	8	CART	0.108	0.108	0.108	0.108	0.108	0.108

Table 1. *Accuracy for each of the classification tests. The best value for each dataset is shown in bold. The dataset number corresponds with that in Table* **??***, but for completeness the arrival distribution, congestion and variables used are also included here. C is the* congestion *variable; R is the* route *variable; and S is the* start time *variable.*

Arrival distribution	*Congested*	*Variables*	*Dataset*	*Method*	*ME*	*MAE*	*MSE*	*MAPE*	*RMSE*
		C,R,S	1	Random Forest	-21	1462	3783000	0.25	1945
		C,R,S	1	CART	-6	**1151**	**2228000**	**0.18**	**1492**
	Yes	C,R,S	1	Linear regression	-2	1156	2249000	0.18	1499
		R,S	2	Random Forest	12	2405	9006000	0.41	3000
		R,S	2	CART	**-1**	1570	4397000	0.24	2097
β		R,S	2	Linear regression	-4	1567	4364000	0.24	2089
		C,R,S	3	Random Forest	0	234	85440	0.22	292
		C,R,S	3	CART	1	194	60670	**0.17**	246
	No	C,R,S	3	Linear regression	-3	**194**	**60310**	**0.17**	**246**
		R,S	4	Random Forest	**0**	337	169800	0.32	412
		R,S	4	CART	-5	341	173000	0.32	416
		R,S	4	Linear regression	-3	339	172000	0.32	415
		C,R,S	5	Random Forest	-2	498	434000	0.14	659
		C,R,S	5	CART	**0**	353	198700	**0.08**	446
	Yes	C,R,S	5	Linear regression	1	**352**	**197200**	**0.08**	**444**
		R,S	6	Random Forest	-4	696	1074000	0.23	1036
		R,S	6	CART	7	591	816700	0.20	904
High-low-high		R,S	6	Linear regression	6	587	811300	0.20	900
		C,R,S	7	Random Forest	-8	587	560100	0.21	749
		C,R,S	7	CART	**-2**	362	204400	**0.11**	452
	No	C,R,S	7	Linear regression	5	**359**	**200400**	**0.11**	**448**
		R,S	8	Random Forest	-2	1084	1696000	0.40	1302
		R,S	8	CART	-5	1068	1621000	0.39	1273
		R,S	8	Linear regression	-4	1066	1614000	0.39	1270

Table 2. *Mean performance for each of the regression tests. Values are in seconds, except for MAPE, which is a fraction. The best value for each dataset is shown in bold.The dataset number corresponds with that in Table* **??***, but for completeness the arrival distribution, congestion and variables used are also included here. C is the* congestion *variable; R is the* route *variable; and S is the* start time *variable.*
The metric abbreviations are: ME = Mean error; MAE = Mean absolute.error; MSE = Mean square.error; MAPE = Mean absolute percentage error; RMSE = Root mean square error