Machine Learning Algorithms for Negotiation Counter-Offer Modeling

Réal Carbonneau

Department of Management Sciences, HEC Montréal
3000, chemin de la Côte-Sainte-Catherine
Montréal (Québec) Canada H3T 2A7
real.carbonneau@hec.ca

Keywords: negotiation; negotiation support systems; machine learning, data mining, neural networks, random forests, support vector machines, opponent modeling, counter-offer modeling.

1 Introduction

With the widespread adoption of computers, the internet, information systems and e-commerce, electronic negotiations have become an important research consideration (G. E. Kersten and S. J. Noronha, 1999). With this emergence of electronic negotiations also comes the possibility of computer assisted and even automated negotiations.

The current research examines the use of data mining with various leading machine learning algorithms to model and simulate the counter offers of an opponent during a negotiation session. The algorithms are compared based on their prediction quality and resource requirements.

2 Problem definition

To have enough observations to learn patterns about an opponent’s behaviour during negotiation, data must be collected. Electronic negotiation systems make this feasible and the current research is based on observations collected by Inspire system (G. Kersten and S. J. Noronha, 1999). The data comes from the Itex/Cypress negotiation case with price, delivery, payment and returns negotiation issues.

The information structure of the model is based on the common denominator of most negotiation offers which is the offer issue levels and the time the offer was made. Because negotiations operate in a time-series fashion, details about the last buy offer and sell offer are also relevant as are some other statistics which outline the negotiation state and progress. In addition to these, a categorical variable indicating if the offer is from a buyer or seller is also included because of its obvious importance. A summary of the inputs (independent) variables and outputs (dependent) variables are presented in Table 1.
3 Algorithms

Several leading machine learning algorithms are considered for modeling the offers made by negotiation opponents, each one of these can learn patterns from a set of data and use these patterns to predict outcomes in new situations (new data). These algorithms are Neural Networks, Random Forests and Support Vector Machines, in addition, Multiple Linear Regression is provided as a baseline reference.

3.1 Neural Networks

A Neural Network is network of weights and transfer functions (usually non-linear), where the weights are adjusted to model complex patterns which related the input variables to the output variables. One general advantage of the Neural Networks over Random Forests and Support Vector Machines is the flexibility with which a machine learning model can be designed. In particular, the Neural Networks can be designed with exactly the number of inputs and outputs required. For the current opponent modeling problem, the neural network has 39 inputs, 4 outputs which fits exactly our modeling problem. The Levenberg-Marquardt with Bayesian Regularization algorithm is used for training the neural networks as an alternative to early stopping to control over-fitting (Foresee and Hagan, 1997; Martin T. Hagan, Demuth, and Beale, 1996; M. T. Hagan and Menhaj, 1994; MacKay, 1992; Marquardt, 1963).

3.2 Random Forests

Random Forest is an algorithm which generates a large set of decision or regression trees with random variations in the set of predictors and the observations selected (Breiman, 2001). The decision or regression trees are grown to their maximum depth, however, as a result of the random variations introduced in each tree, the set of trees does not usually overfit the data and the error estimates of the model are representative of out-of-sample performance. Another advantage of Random Forests is that the algorithm can handle a very large number of input (dependant) variables without degradation of the performance, whereas many other algorithms suffer from the curse of dimensionality. The Random Forest implementation developed for the R-Project (Liaw and Wiener, 2002; R Development Core Team, 2006) is used for this research.

3.3 Support Vector Machines

Support vector machines (SVM) are based on the structural risk minimization principle from statistical learning theory (V. Vapnik, Golowich, and Smola, 1997; V. N. Vapnik, 1995) as opposed to the empirical risk minimization principle on which artificial neural networks (ANN) and multiple linear regression (MLR), to name a few, are based. The objective of structural risk minimization is to
reduce the true error on an unseen and randomly selected test example as opposed to ANN and MLR, which minimize the error for the currently seen examples.

4 Preliminary Results

Although extensive modeling is still ongoing, some preliminary results are available. The best models so far have been from the Random Forest algorithm in classification mode, but regression model also provides very good performance. The Neural Network models also perform well and are close behind the Random Forest, however, the processing time (3 Days) and memory requirements for the Random Forests (50GB) are much heavier than the Neural Networks (3 Hours and 400 MB). Both the Neural Network models and the Random Forest outperform the Linear models by about 10% (Table 2).

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Price</th>
<th>Delivery</th>
<th>Payment</th>
<th>Returns</th>
<th>Avg</th>
<th>Time</th>
<th>Memory</th>
</tr>
</thead>
<tbody>
<tr>
<td>NN LM BR BR 10 Neurons</td>
<td>0.4231</td>
<td>0.3613</td>
<td>0.2987</td>
<td>0.3273</td>
<td>0.3526</td>
<td>30 Mins</td>
<td>350 MB</td>
</tr>
<tr>
<td>Random Forest Class. 5000 trees</td>
<td>0.3970</td>
<td>0.3693</td>
<td>0.2845</td>
<td>0.3241</td>
<td>0.3437</td>
<td>3 Days</td>
<td>50 GB</td>
</tr>
<tr>
<td>Random Forest Regr. 5000 trees</td>
<td>0.3938</td>
<td>0.3716</td>
<td>0.3027</td>
<td>0.3431</td>
<td>0.3528</td>
<td>5 Days</td>
<td>50 GB</td>
</tr>
<tr>
<td>Multiple Linear Regression</td>
<td>0.4873</td>
<td>0.3867</td>
<td>0.3074</td>
<td>0.3487</td>
<td>0.3825</td>
<td>1 Min</td>
<td>100 MB</td>
</tr>
</tbody>
</table>

Although the processing requirements for Random Forest are many times greater than Neural Network training, they also offer the advantage of more constant results and better protection against over-fitting. Currently, Support Vector Machines are providing worse performance than the baseline linear model, therefore, the results are not presented because of a potential algorithm problem. At this stage in the research, the Random Forest algorithm seems to offer the best performance for modeling negotiation counter offers even though the processing requirements are much higher than neural networks.

References