Autonomic Forecasting Method Selection: Examination and Ways Ahead

BASED ON: PAPER BY MARWIN ZUFLE, UNIVERSITY OF WURZBERG
Problem Description

- No general forecasting method that performs best in all scenarios
- Gap between design time and runtime scenarios for adaptive systems
- The uncertainty resulting from the gap decreases the possibility that a forecasting method chosen at design time would perform well at runtime
- Need to design recommendation systems which automatically choose the forecasting method at runtime
In 2009, Wang et al. proposed clustering and rule induction algorithms to generate categorical and quantitative rules based on a large variety of time series features.

Characteristics of time series data like trend, seasonality, periodicity, serial correlation, skewness, kurtosis, non-linearity, self-similarity and chaos are measured.

Authors introduced nine time series characteristics that are assumed to have a relation to the performance of four forecasting methods, i.e., ARIMA, ETS, ANN, and random walk.

Class label is set to 1 for the best forecasting method for each time series. All other forecasting methods receive the class label 0 for the respective time series.

These labels are used as prediction class and the time series characteristics are used as meta-level attributes. Then, the C4.5 (decision tree technique) algorithm is applied with this data on each forecasting method to generate quantitative recommendation rules.

This is a static rule learning approach.
Knowledge Acquisition Mode

- Time series examples
- Time series forecasting methods evaluation
- Base-level methods (prediction results)
- Meta-level data set
- Rule generation
- Recommendation rules (knowledge base)
- Meta-level attributes
Binary Classification with Oversampling

Schematic process of the rule generation approach. First, oversampling is applied to the class label vectors and the matrix of characteristics $T_1$ to $T_4$ are then used to train a random forest model $M_1$ to $M_4$ for binary classification forecasting method in the recommendation system. The prediction of those models for a validation time series $V$ results in probabilities for the class labels 0 and 1 for each forecasting method. The forecasting method with the highest probability of class label 1 is returned.
Recommendation-based Ensemble Forecasting

Combination of ensemble forecasting and forecasting method recommendation.

Applies linear regression for automatically adjusting the weights for a linear combination of forecasting methods.

Each weight $\vartheta_i$ is estimated by a linear regression of the characteristics of the previously seen time series.

We interpret this output as the probability of how good the forecasting method is suitable for the time series.

We activate a weight if it fulfills both criteria: the weight $\vartheta_i$ must be (i) greater or equal to the mean of the weights $\vartheta$ and (ii) greater or equal to the share $\alpha$ of the maximum weight $\vartheta^*$. 

\[
Y = \frac{1}{\vartheta} \sum_{i \in \{A,E,N,R\}} Y^i \cdot \Gamma(\vartheta^i) \quad \text{with} \quad \vartheta = \sum_{i \in \{A,E,N,R\}} \Gamma(\vartheta^i)
\]

\[
\Gamma(\vartheta^i) = \vartheta^i \cdot \left(1 - \max(\text{sign}(\max(\vartheta, \alpha \cdot \hat{\vartheta}) - \vartheta^i), 0)\right)
\]

\[
\vartheta = \text{mean}(\vartheta^i), \quad \text{and} \quad \hat{\vartheta} = \max(\vartheta^i) \quad \text{for} \quad i \in \{A,E,N,R\}
\]
Both the binary classification with oversampling approach as well as the recommendation-based ensemble forecasting approach can be applied for runtime performance decisions in autonomous systems. The rules and weights can be used for new and unseen time series at runtime since applying the recommendation is very fast. Moreover, these new time series can then be used to dynamically re-learn the recommendation rules and weights. As we rely on a random forest-based approach for rule generation, the dynamic re-learning of rules is feasible with a small overhead at runtime.
Results

The forecast accuracy degradation indicates how much lower the associated method performs compared to the best method.

**TABLE VI**

Degradation in accuracy for all approaches averaged over all 100 random splits.

<table>
<thead>
<tr>
<th>Data set</th>
<th>Wang <em>et al.</em></th>
<th>ARIMA</th>
<th>ETS</th>
<th>ANN</th>
<th>Random walk</th>
<th>Bin. Class. + Oversampling</th>
<th>Recom. + Ensemble</th>
</tr>
</thead>
<tbody>
<tr>
<td>CR multi-step</td>
<td>69.8%</td>
<td>59.9%</td>
<td>46.2%</td>
<td>91.0%</td>
<td>83.6%</td>
<td>42.3%</td>
<td>40.6%</td>
</tr>
<tr>
<td>CR multi-step</td>
<td>404.9%</td>
<td>98.0%</td>
<td>345.6%</td>
<td>386.3%</td>
<td>300.9%</td>
<td>82.1%</td>
<td>104.9%</td>
</tr>
</tbody>
</table>
Thank You