An Empirical Comparison of Methods to support QoS-aware Service Selection

B. Cavallo[1 2] M. Di Penta[2] G. Canfora[2]

[1]Department of Constructions and Mathematical Methods in Architecture
University of Naples, Federico II, Italy

[2]Department of Engineering-RCOST
University of Sannio, Italy

PESOS '10, May 1-2, 2010, Cape Town, South Africa

Outline

Outline

Late binding is a mechanism which allows to bind a request coming from a service composition—hereby referred as *abstract service*—to one of the (possibly) multiple *concrete services* available and able to satisfy the specific request.

Example

a flight booking request can be forwarded to services belonging to different airlines

Example

an e-book search request can be forwarded to services belonging to different e-libraries

Late binding is a mechanism which allows to bind a request coming from a service composition—hereby referred as *abstract service*—to one of the (possibly) multiple *concrete services* available and able to satisfy the specific request.

Example

a flight booking request can be forwarded to services belonging to different airlines

Example

an e-book search request can be forwarded to services belonging to different e-libraries

Late binding is a mechanism which allows to bind a request coming from a service composition—hereby referred as *abstract service*—to one of the (possibly) multiple *concrete services* available and able to satisfy the specific request.

Example

a flight booking request can be forwarded to services belonging to different airlines

Example

an e-book search request can be forwarded to services belonging to different e-libraries

Outline

QoS (Quality of Service)-awareness implies the enactment of different mechanisms, for which approaches and tools have been developed in recent and past years, such as:

- monitoring mechanisms to collect QoS and functional information about service invocations, and to trigger recovery actions whenever needed
- approaches to estimate the QoS of a service composition given the QoS of services that participate in the composition
- approaches to enact dynamic binding and to determine the (near) optimal set of bindings for a service composition

QoS (Quality of Service)-awareness implies the enactment of different mechanisms, for which approaches and tools have been developed in recent and past years, such as:

- monitoring mechanisms to collect QoS and functional information about service invocations, and to trigger recovery actions whenever needed
- approaches to estimate the QoS of a service composition given the QoS of services that participate in the composition
- approaches to enact dynamic binding and to determine the (near) optimal set of bindings for a service composition

QoS (Quality of Service)-awareness implies the enactment of different mechanisms, for which approaches and tools have been developed in recent and past years, such as:

- monitoring mechanisms to collect QoS and functional information about service invocations, and to trigger recovery actions whenever needed
- approaches to estimate the QoS of a service composition given the QoS of services that participate in the composition
- approaches to enact dynamic binding and to determine the (near) optimal set of bindings for a service composition

Outline

- the paper reports results from an empirical study aimed at comparing different approaches for QoS prediction
- the study has been performed upon QoS data—we restrict our attention to response time—collected by invoking and monitoring 10 real services for 4 months every hour
- What is the prediction error produced by the different approaches?
- To what the extant approaches can be used to forecast QoS violations?

- the paper reports results from an empirical study aimed at comparing different approaches for QoS prediction
- the study has been performed upon QoS data—we restrict our attention to response time—collected by invoking and monitoring 10 real services for 4 months every hour
- What is the prediction error produced by the different approaches?
- To what the extant approaches can be used to forecast QoS violations?

- the paper reports results from an empirical study aimed at comparing different approaches for QoS prediction
- the study has been performed upon QoS data—we restrict our attention to response time—collected by invoking and monitoring 10 real services for 4 months every hour
- What is the prediction error produced by the different approaches?
- To what the extant approaches can be used to forecast QoS violations?

- the paper reports results from an empirical study aimed at comparing different approaches for QoS prediction
- the study has been performed upon QoS data—we restrict our attention to response time—collected by invoking and monitoring 10 real services for 4 months every hour
- What is the prediction error produced by the different approaches?
- To what the extant approaches can be used to forecast QoS violations?

Outline

Let X be the attribute to use for QoS-aware selection:

$$\hat{X}(t_0), \hat{X}(t_1), \ldots, \hat{X}(t_k), \hat{X}(t_{k+1})$$

Training data points to build the prediction model

- *once n predictions have been performed, the training set is augmented with the next n actual values of the time series, the model is estimated again, and the subsequent n values are then predicted*
- *we focus on response time and consider as initial training set the first 500 values and predictions at 1-step ahead—i.e., how the service will respond one hour from now*

Let X be the attribute to use for QoS-aware selection:

$$\hat{X}(t_0), \hat{X}(t_1), \ldots, \hat{X}(t_k), \hat{X}(t_{k+1}), \hat{X}(t_{k+2})$$

Training data points to build the prediction model

- *once n predictions have been performed, the training set is augmented with the next n actual values of the time series, the model is estimated again, and the subsequent n values are then predicted*
- *we focus on response time and consider as initial training set the first 500 values and predictions at 1-step ahead—i.e., how the service will respond one hour from now*

Let X be the attribute to use for QoS-aware selection:

$$\hat{X}(t_0), \hat{X}(t_1), \ldots, \hat{X}(t_k), \hat{X}(t_{k+1}), \hat{X}(t_{k+2}), \hat{X}(t_{k+3})$$

Training data points to build the prediction model

- *once n predictions have been performed, the training set is augmented with the next n actual values of the time series, the model is estimated again, and the subsequent n values are then predicted*
- *we focus on response time and consider as initial training set the first 500 values and predictions at 1-step ahead—i.e., how the service will respond one hour from now*

Average of past values

The forecasting of n-steps ahead at time t_k is based on the average of the previously observed $k+1$ values:

$$X(t_{k+n}) = \sum_{i=0}^{k} \frac{\hat{X}(t_i)}{k+1}$$

where $X(t_{k+n})$ is the predicted value at time t_{k+n}, while $\hat{X}(t_i)$ is the observed value at time t_i.

Current value

This prediction model is very simple, as it just uses the last observed QoS value:

$$X(t_{k+n}) = \hat{X}(t_k)$$

Linear model

This approach builds a linear regression model, using the minimum least squares method, over the previously observed $k + 1$ *values, and builds an equation*

$$X(t_i) = a \cdot t_i + b$$

which can be used to predict future values of the time series.

ARMA(p, q) (Autoregressive Moving Average)

$MA(q)$ explains the present as the mixture of q random impulses, while an $AR(p)$ process builds the present in terms of the past p events. By combining MA of order p and AR processes of order q, we obtain an ARMA process of order (p, q), defined as follows:

$$X_t = \alpha_1 X_{t-1} + \ldots + \alpha_p X_{t-p} + Z_t + \beta_1 Z_{t-1} + \ldots + \beta_q Z_{t-q}$$

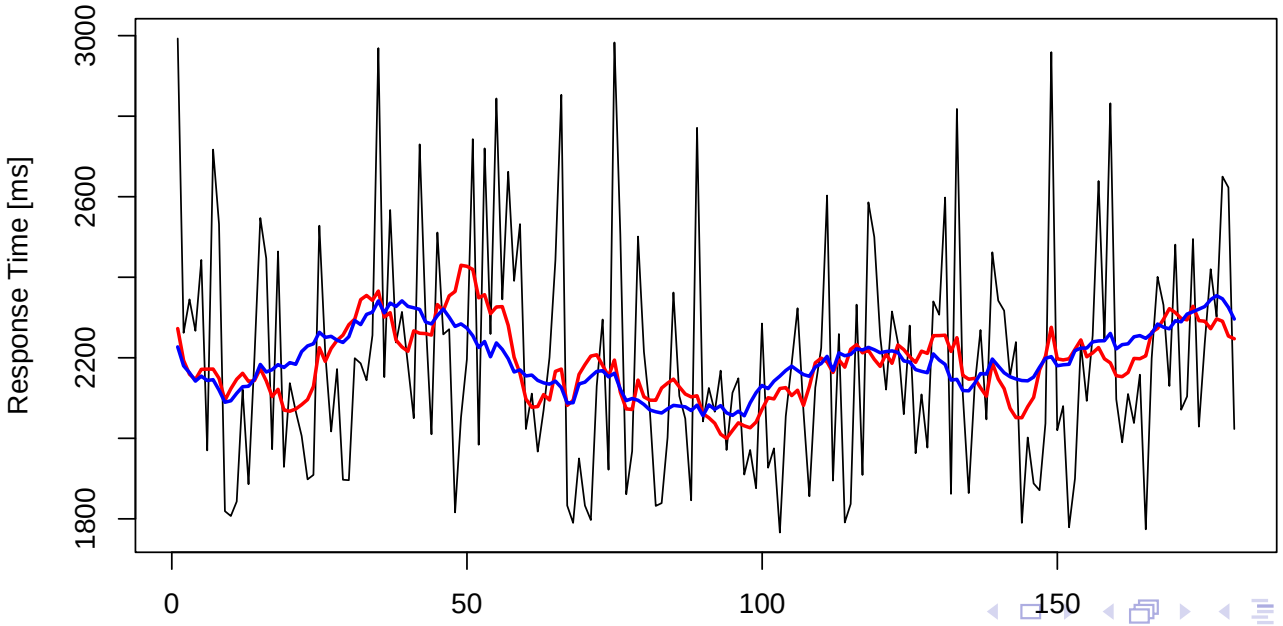
where $X(t)$ is the original series and $Z(t)$ is a series of random pulses which are assumed to follow the normal probability distribution.

ARIMA(p, d, q) (Auto-Regressive Integrated Moving Average)

A generalization of ARMA processes to deal with the modeling of non-stationary time series.

Smoothed time series

To avoid the training data set being polluted by outliers and noise, we smooth it using a kernel smoothing function.
A kernel smoother is a statistical technique that allows to estimate a real function from its observations.



Outline

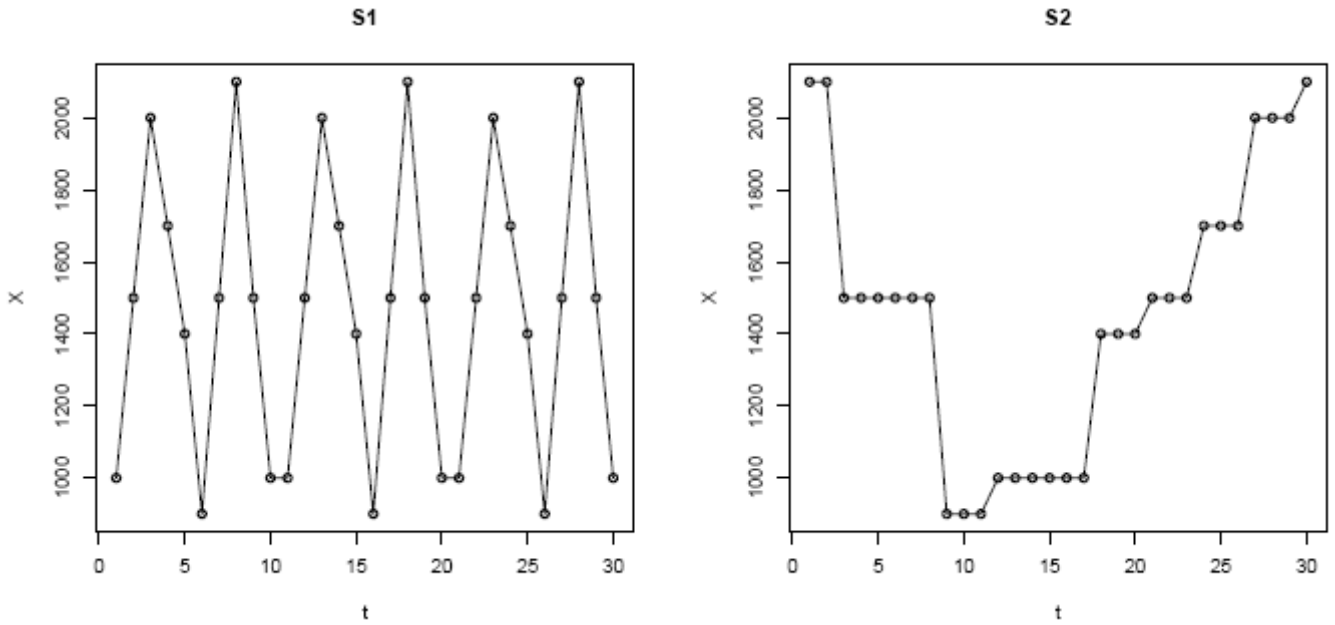
1. *Identifying the presence of trends*. If the time series is not stationary an ARIMA time series need to be used.
   We use the Augmented Dickey-Fuller (ADF) test to check whether the time series contains a trend;
2. *Identifying seasonal (periodic) component of the series*. This is done by analyzing the spectral decomposition of the time series;
3. *Model identification*. The observed time series is analyzed to select an ARIMA(p, d, q) process that appears to be the most appropriate;
4. *Estimation.* The actual time series is modeled using the ARIMA(p, d, q) process previously defined.

1. *Identifying the presence of trends*. If the time series is not stationary an ARIMA time series need to be used.
   We use the Augmented Dickey-Fuller (ADF) test to check whether the time series contains a trend;
2. *Identifying seasonal (periodic) component of the series*. This is done by analyzing the spectral decomposition of the time series;
3. *Model identification*. The observed time series is analyzed to select an ARIMA(p, d, q) process that appears to be the most appropriate;
4. *Estimation.* The actual time series is modeled using the ARIMA(p, d, q) process previously defined.

1. *Identifying the presence of trends*. If the time series is not stationary an ARIMA time series need to be used.
   We use the Augmented Dickey-Fuller (ADF) test to check whether the time series contains a trend;
2. *Identifying seasonal (periodic) component of the series*. This is done by analyzing the spectral decomposition of the time series;
3. *Model identification*. The observed time series is analyzed to select an ARIMA(p, d, q) process that appears to be the most appropriate;
4. *Estimation.* The actual time series is modeled using the ARIMA(p, d, q) process previously defined.

1. *Identifying the presence of trends*. If the time series is not stationary an ARIMA time series need to be used.
   We use the Augmented Dickey-Fuller (ADF) test to check whether the time series contains a trend;
2. *Identifying seasonal (periodic) component of the series*. This is done by analyzing the spectral decomposition of the time series;
3. *Model identification*. The observed time series is analyzed to select an ARIMA(p, d, q) process that appears to be the most appropriate;
4. *Estimation.* The actual time series is modeled using the ARIMA(p, d, q) process previously defined.

Outline



Goal: to select the service that, with higher likelihood, will meet a QoS constraint in future invocations

- $X \leq 1600$. S_1, S_2 meet the constraint in 70% of the cases
- simple descriptive statistics would not allow us to determine which service is better
  - S_1, S_2: the minimum, maximum and average response times are 900, 2100 and 1460

Outline

- The goal of this study is to compare the capabilities of different approaches for QoS prediction, in the context of dynamic QoS-aware service selection for SOA (Service Oriented Architectures).
- The focus is on selecting the services that provide the best QoS, while avoiding SLA (Service Level Agreement) constraint violations.
- The context consists of monitored QoS data collected by invoking 10 real services every hour for about four months.

RQ1: *What is the prediction error produced by the different approaches?*

To address **RQ1**, *for each performed prediction, we compute the relative error defined as:*

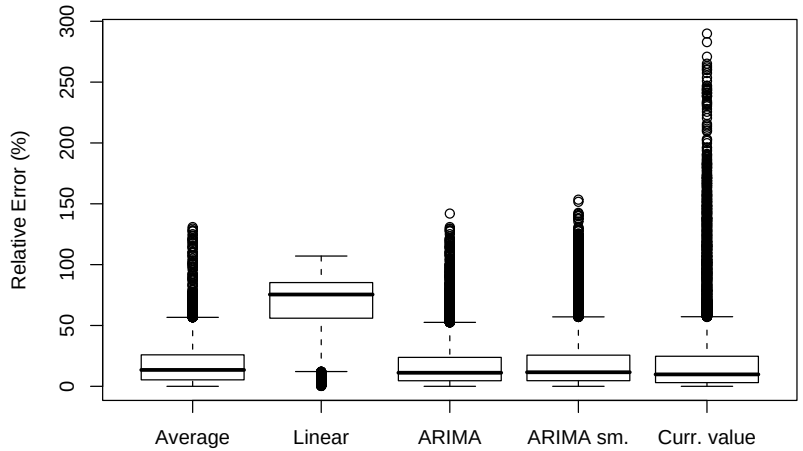
$$e(t_i) = \frac{|x(t_i) - \hat{x}(t_i)|}{\hat{x}(t_i)}$$

and we analyze it using boxplots and descriptive statistics.

RQ2: *To what the extant approaches can be used to forecast QoS violations?*

- *to address* **RQ2***, we check, for two different constraints whether the predicted QoS value violates or not the constraint;*
- *we compute the percentage of correctly predicted QoS violations*

Outline



- the linear model produces the most high prediction error
- ARIMA (ARIMA $(p, 0, q)$) error is lower than those of current value and average value
- ARIMA performs better without smoothing the time series
- the current value model produces a high number of outliers, exhibiting a very high prediction error, as confirmed by the high standard deviation.

Outline

Service	Constraint 1			Constraint 2		
	ARIMA	ARIMA sm.	Curr. value	ARIMA	ARIMA sm.	Curr. value
S_1	11	25	29	6	12	25
S_2	3	17	15	100	91	33
S_3	0	0	20	0	0	15
S_4	69	40	76	41	17	69
S_5	7	7	31	0	1	18
S_6	2	21	36	1	8	25
S_7	0	0	44	0	0	21
S_8	21	20	34	8	8	30
S_9	51	55	47	15	18	30
S_{10}	30	35	38	17	21	34

- the average percentages of correct predictions
  - C1 19%, 22%, 37%
  - C2 19%, 18%, 30%
- there is a higher percentage of correctly predicted violations for the current value model
- there is no significant difference between ARIMA and ARIMA smoothed

Results show that:

- the most useful model for forecasting QoS violations is the current value, although ARIMA is still able to perform a reasonably good number of predictions
- the reason why the current value outperforms ARIMA is, in our understanding, due to the relatively low capability ARIMA has to predict values strongly deviating from the average, and thus violations
- guaranteeing a good prediction of SLA violations is still a challenging issue, thus further models able to better deal with this issue should be investigated

Results show that:

- the most useful model for forecasting QoS violations is the current value, although ARIMA is still able to perform a reasonably good number of predictions
- the reason why the current value outperforms ARIMA is, in our understanding, due to the relatively low capability ARIMA has to predict values strongly deviating from the average, and thus violations
- guaranteeing a good prediction of SLA violations is still a challenging issue, thus further models able to better deal with this issue should be investigated

Results show that:

- the most useful model for forecasting QoS violations is the current value, although ARIMA is still able to perform a reasonably good number of predictions
- the reason why the current value outperforms ARIMA is, in our understanding, due to the relatively low capability ARIMA has to predict values strongly deviating from the average, and thus violations
- guaranteeing a good prediction of SLA violations is still a challenging issue, thus further models able to better deal with this issue should be investigated

- This paper reports an empirical study aimed at comparing different QoS prediction models on time series of response time collected by monitoring invocations of 10 services for 4 months
- Results of the study indicate that ARIMA time series forecasting constitutes a good compromise in terms of achieving a low prediction error and being capable to predict SLA violations
- Simpler models such as the average of the previously observed QoS also exhibit a low prediction error, but cannot be used to predict SLA violations
- Models based on the current value, instead, exhibit high prediction of SLA violations
- The usage of smoothing does not help to improve the predictions

- This paper reports an empirical study aimed at comparing different QoS prediction models on time series of response time collected by monitoring invocations of 10 services for 4 months
- Results of the study indicate that ARIMA time series forecasting constitutes a good compromise in terms of achieving a low prediction error and being capable to predict SLA violations
- Simpler models such as the average of the previously observed QoS also exhibit a low prediction error, but cannot be used to predict SLA violations
- Models based on the current value, instead, exhibit high prediction of SLA violations
- The usage of smoothing does not help to improve the predictions

- This paper reports an empirical study aimed at comparing different QoS prediction models on time series of response time collected by monitoring invocations of 10 services for 4 months
- Results of the study indicate that ARIMA time series forecasting constitutes a good compromise in terms of achieving a low prediction error and being capable to predict SLA violations
- Simpler models such as the average of the previously observed QoS also exhibit a low prediction error, but cannot be used to predict SLA violations
- Models based on the current value, instead, exhibit high prediction of SLA violations
- The usage of smoothing does not help to improve the predictions

- This paper reports an empirical study aimed at comparing different QoS prediction models on time series of response time collected by monitoring invocations of 10 services for 4 months
- Results of the study indicate that ARIMA time series forecasting constitutes a good compromise in terms of achieving a low prediction error and being capable to predict SLA violations
- Simpler models such as the average of the previously observed QoS also exhibit a low prediction error, but cannot be used to predict SLA violations
- Models based on the current value, instead, exhibit high prediction of SLA violations
- The usage of smoothing does not help to improve the predictions

- This paper reports an empirical study aimed at comparing different QoS prediction models on time series of response time collected by monitoring invocations of 10 services for 4 months
- Results of the study indicate that ARIMA time series forecasting constitutes a good compromise in terms of achieving a low prediction error and being capable to predict SLA violations
- Simpler models such as the average of the previously observed QoS also exhibit a low prediction error, but cannot be used to predict SLA violations
- Models based on the current value, instead, exhibit high prediction of SLA violations
- The usage of smoothing does not help to improve the predictions

Work-in-progress aims at investigating:

- models accounting for periodicity of the time series which was not observed in the current data set
  - response times decrease during weekends
  - response times increase during peak hours
- other forecasting models, which could be used to better predict SLA violations

THANKS FOR YOUR ATTENTION