Predicting Quality of Service (QoS) Parameters using Extreme Learning Machines with Various Kernel Methods

Lov Kumar\textsuperscript{1}  Santanu Kumar Rath\textsuperscript{1}  Ashish Sureka\textsuperscript{2}

\textsuperscript{1}NIT Rourkela, India
Email: lovkumar505@gmail.com

\textsuperscript{2}ABB India, India
Email: ashish.sureka@in.abb.com

QuASoC 2016
Table of Contents

1. Research Motivation and Aim
2. Related Work
3. Research Framework and Experimental Dataset
   - Research Framework
   - Experimental Dataset
4. Variables - Empirical Analysis
   - Dependent Variable
   - Predictor Variables
   - Code Metrics - Correlation Analysis
5. Feature Extraction and Selection
   - Principal Component Analysis (PCA)
   - Rough Set Analysis (RSA)
6. Extreme Learning Machines (ELMs)
7. Conclusion
   - Conclusion - I
   - Conclusion - II
8. References

Lov Kumar, Santanu Kumar Rath, Ashish Sureka
Predicting Quality of Service (QoS) Parameters using Extreme L
Web services: distributed web application components, can be implemented in different languages, represented by interfaces and communicate using open protocols [6][14]

Applications and business solutions using web services expect high **Quality of Service (QoS)** as their application is dependent on the service

**QoS Attributes:** response time, availability, throughput, reliability, modularity, testability and interoperability

**Source code metrics:** Object oriented metrics (Chidamber and Kemerer), Baski & Misra and Harry M. Sneed metrics
Research Objectives

[1] To investigate the **correlation** between QoS attributes and source code metrics

[2] To study the correlation between **15** web service quality attributes and **37** source code metrics

[3] To build **machine learning based predictive models** for estimating the quality of a given service based on the computed source code metrics

[4] Examine the extent to which feature selection techniques such as Principal Component Analysis (PCA) and Rough Set Analysis (RSA) can be used for **dimensionality reduction** and filter irrelevant features.
Coscia et al. [4] investigate the potential of obtaining more maintainable services by exploiting Object-Oriented metrics (OO) values from the source code implementing services [4].

Coscia and Crasso et al. [5] present a statistical correlation analysis demonstrating that classic software engineering metrics (such as WMC, CBO, RFC, CAM, TPC, APC and LCOM) can be used to predict the most relevant quality attributes of WSDL documents [5].
Mateos et al. [13]

Mateos et al. found that there is a high correlation between well-known object-oriented metrics taken in the code implementing services and the occurrences of anti-patterns in their WSDLs [13].

Kumar et al. [12]

Kumar et al. use different object-oriented software metrics and Support Vector Machines with different type of kernels for predicting maintainability of services [12].

Olatunji et al. [15]

Olatunji et al. develop an extreme learning machine (ELM) main-
# Table of Contents

1. Research Motivation and Aim
2. Related Work
3. Research Framework and Experimental Dataset
   - Research Framework
   - Experimental Dataset
4. Variables - Empirical Analysis
   - Dependent Variable
   - Predictor Variables
   - Code Metrics - Correlation Analysis
5. Feature Extraction and Selection
   - Principal Component Analysis (PCA)
   - Rough Set Analysis (RSA)
6. Extreme Learning Machines (ELMs)
7. Conclusion
   - Conclusion - I
   - Conclusion - II
8. References

---

**Lov Kumar, Santanu Kumar Rath, Ashish Sureka**

*Predicting Quality of Service (QoS) Parameters using Extreme L*
Multi-Step Methodology

1. Web Services
   - WSDL to Java
   - Source Code
2. Quality of Service (QoS) Parameters
   - Baski and Misra
   - Chidamber and Kemerer
   - Henry M. Sneed
3. Rough Set Analysis (RSA)
   - PCA
   - Feature Selection
   - Linear Kernel
   - Polynomial Kernel
   - RBF Kernel
4. Performance Evaluation
   - Statistical Tests
   - Model Comparison

Lov Kumar, Santanu Kumar Rath, Ashish Sureka
Predicting Quality of Service (QoS) Parameters using Extreme Learning Machines with Various Kernel Methods
Framework and Methodology

[1] We compute **37 source code metrics** belonging to 3 different metrics suite

[2] We apply **2 different feature selection methodology** (Rough Set Analysis and Principal Component Analysis) for the purpose of dimensionality reduction and removing irrelevant features

[3] We apply **Extreme Learning Machines (ELM)** with three different kernel functions (linear, polynomial and RBF)

[4] We create 6 sets of metrics suite, 2 feature selection techniques and **3 kernel functions** and evaluate the performance of all the combinations
Table of Contents

1. Research Motivation and Aim
2. Related Work
3. Research Framework and Experimental Dataset
   - Research Framework
   - Experimental Dataset
4. Variables - Empirical Analysis
   - Dependent Variable
   - Predictor Variables
   - Code Metrics - Correlation Analysis
5. Feature Extraction and Selection
   - Principal Component Analysis (PCA)
   - Rough Set Analysis (RSA)
6. Extreme Learning Machines (ELMs)
7. Conclusion
   - Conclusion - I
   - Conclusion - II
8. References

Lov Kumar, Santanu Kumar Rath, Ashish Sureka
Predicting Quality of Service (QoS) Parameters using Extreme L
QWS Dataset

[1] We use a subset of QWS Dataset\(^a\)

[2] The QWS Dataset provided by Al-Masri et al. includes a sets of 2507 Web services and their 9 QWS parameters

[3] We apply the Baski and Misra metrics suite tool on the 524 WSDL files and obtained successful parsing for 200 files

[4] We finally use 200 Web services for the experiments presented in this paper

\(^a\)http://www.uoguelph.ca/~qmahmoud/qws/
Scatter Plot for the Number of Java Files for the 200 WSDL Files in Experimental Dataset

Lov Kumar, Santanu Kumar Rath, Ashish Sureka

Predicting Quality of Service (QoS) Parameters using Extreme Learning Machines with Various Kernel Methods
### Descriptive Statistics of QoS Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Min</th>
<th>Max</th>
<th>Mean</th>
<th>Median</th>
<th>Std Dev</th>
<th>Skewness</th>
<th>Kurtosis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Response Time</td>
<td>57.00</td>
<td>1664.62</td>
<td>325.11</td>
<td>252.20</td>
<td>289.33</td>
<td>3.15</td>
<td>13.12</td>
</tr>
<tr>
<td>Availability</td>
<td>13.00</td>
<td>100.00</td>
<td>86.65</td>
<td>89.00</td>
<td>12.57</td>
<td>-2.63</td>
<td>12.06</td>
</tr>
<tr>
<td>Throughput</td>
<td>0.20</td>
<td>36.90</td>
<td>7.04</td>
<td>4.00</td>
<td>6.94</td>
<td>1.57</td>
<td>5.79</td>
</tr>
<tr>
<td>Successability</td>
<td>14.00</td>
<td>100.00</td>
<td>90.19</td>
<td>96.00</td>
<td>13.61</td>
<td>-2.57</td>
<td>10.81</td>
</tr>
<tr>
<td>Reliability</td>
<td>33.00</td>
<td>83.00</td>
<td>66.64</td>
<td>73.00</td>
<td>9.61</td>
<td>-0.60</td>
<td>2.96</td>
</tr>
<tr>
<td>Compliance</td>
<td>67.00</td>
<td>100.00</td>
<td>92.19</td>
<td>100.00</td>
<td>9.78</td>
<td>-0.90</td>
<td>2.61</td>
</tr>
<tr>
<td>Best Practices</td>
<td>57.00</td>
<td>93.00</td>
<td>78.81</td>
<td>82.00</td>
<td>7.70</td>
<td>-0.68</td>
<td>2.67</td>
</tr>
<tr>
<td>Latency</td>
<td>0.74</td>
<td>1337.00</td>
<td>42.81</td>
<td>12.20</td>
<td>106.23</td>
<td>9.56</td>
<td>112.68</td>
</tr>
<tr>
<td>Documentation</td>
<td>1.00</td>
<td>96.00</td>
<td>29.37</td>
<td>32.00</td>
<td>26.97</td>
<td>1.06</td>
<td>3.31</td>
</tr>
</tbody>
</table>

Substantial variation or dispersion in the parameter values across 200 web services
Table of Contents

1 Research Motivation and Aim
2 Related Work
3 Research Framework and Experimental Dataset
   • Research Framework
   • Experimental Dataset
4 Variables - Empirical Analysis
   • Dependent Variable
   • Predictor Variables
   • Code Metrics - Correlation Analysis
5 Feature Extraction and Selection
   • Principal Component Analysis (PCA)
   • Rough Set Analysis (RSA)
6 Extreme Learning Machines (ELMs)
7 Conclusion
   • Conclusion - I
   • Conclusion - II
8 References

Lov Kumar, Santanu Kumar Rath, Ashish Sureka
Predicting Quality of Service (QoS) Parameters using Extreme Learning Machines
### Chidamber and Kemerer Metrics

<table>
<thead>
<tr>
<th>Metrics</th>
<th>Min</th>
<th>Max</th>
<th>Mean</th>
<th>Median</th>
<th>Std Dev</th>
<th>Skewness</th>
<th>Kurtosis</th>
</tr>
</thead>
<tbody>
<tr>
<td>WMC</td>
<td>9.48</td>
<td>13.57</td>
<td>11.01</td>
<td>10.96</td>
<td>0.48</td>
<td>0.81</td>
<td>6.54</td>
</tr>
<tr>
<td>DIT</td>
<td>0.87</td>
<td>1.02</td>
<td>0.98</td>
<td>0.98</td>
<td>0.02</td>
<td>-2.07</td>
<td>9.72</td>
</tr>
<tr>
<td>NOC</td>
<td>0.00</td>
<td>0.13</td>
<td>0.01</td>
<td>0.01</td>
<td>0.02</td>
<td>2.64</td>
<td>12.55</td>
</tr>
<tr>
<td>CBO</td>
<td>4.10</td>
<td>12.55</td>
<td>10.70</td>
<td>11.01</td>
<td>1.33</td>
<td>-1.15</td>
<td>5.18</td>
</tr>
<tr>
<td>RFC</td>
<td>12.78</td>
<td>44.55</td>
<td>40.35</td>
<td>41.48</td>
<td>4.13</td>
<td>-3.13</td>
<td>15.15</td>
</tr>
<tr>
<td>LCOM</td>
<td>74.03</td>
<td>405.49</td>
<td>120.94</td>
<td>108.67</td>
<td>45.70</td>
<td>2.99</td>
<td>13.96</td>
</tr>
<tr>
<td>Ca</td>
<td>0.64</td>
<td>3.92</td>
<td>2.91</td>
<td>2.99</td>
<td>0.62</td>
<td>-0.50</td>
<td>2.72</td>
</tr>
<tr>
<td>Ce</td>
<td>3.49</td>
<td>9.50</td>
<td>8.24</td>
<td>8.37</td>
<td>0.90</td>
<td>-1.30</td>
<td>6.01</td>
</tr>
<tr>
<td>NPM</td>
<td>4.88</td>
<td>9.27</td>
<td>6.55</td>
<td>6.47</td>
<td>0.48</td>
<td>1.07</td>
<td>7.90</td>
</tr>
<tr>
<td>LCOM3</td>
<td>1.18</td>
<td>1.50</td>
<td>1.32</td>
<td>1.31</td>
<td>0.06</td>
<td>0.27</td>
<td>2.75</td>
</tr>
<tr>
<td>LCO</td>
<td>76.14</td>
<td>493.64</td>
<td>399.18</td>
<td>411.50</td>
<td>54.03</td>
<td>-2.61</td>
<td>11.71</td>
</tr>
<tr>
<td>DAM</td>
<td>0.21</td>
<td>0.45</td>
<td>0.37</td>
<td>0.37</td>
<td>0.04</td>
<td>-0.45</td>
<td>4.53</td>
</tr>
<tr>
<td>MOA</td>
<td>0.02</td>
<td>2.28</td>
<td>0.60</td>
<td>0.53</td>
<td>0.28</td>
<td>2.00</td>
<td>11.03</td>
</tr>
<tr>
<td>MFA</td>
<td>0.00</td>
<td>0.02</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>1.92</td>
<td>8.43</td>
</tr>
<tr>
<td>CAM</td>
<td>0.39</td>
<td>0.43</td>
<td>0.40</td>
<td>0.40</td>
<td>0.01</td>
<td>0.22</td>
<td>4.54</td>
</tr>
<tr>
<td>IC</td>
<td>0.00</td>
<td>0.05</td>
<td>0.01</td>
<td>0.01</td>
<td>0.01</td>
<td>0.79</td>
<td>3.68</td>
</tr>
<tr>
<td>CBM</td>
<td>0.00</td>
<td>0.05</td>
<td>0.01</td>
<td>0.01</td>
<td>0.01</td>
<td>0.79</td>
<td>3.68</td>
</tr>
<tr>
<td>AMC</td>
<td>7.68</td>
<td>82.86</td>
<td>61.94</td>
<td>64.37</td>
<td>10.75</td>
<td>-1.69</td>
<td>6.97</td>
</tr>
</tbody>
</table>

Lov Kumar, Santanu Kumar Rath, Ashish Sureka

Predicting Quality of Service (QoS) Parameters using Extreme Learning Machines with Various Kernel Methods
### Harry M. Sneed’s Metrics Suite

<table>
<thead>
<tr>
<th>Metrics</th>
<th>Min</th>
<th>Max</th>
<th>Mean</th>
<th>Median</th>
<th>Std Dev</th>
<th>Skewness</th>
<th>Kurtosis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Complexity</td>
<td>0.10</td>
<td>0.81</td>
<td>0.28</td>
<td>0.27</td>
<td>0.17</td>
<td>0.60</td>
<td>2.59</td>
</tr>
<tr>
<td>Relation Complexity</td>
<td>0.10</td>
<td>0.90</td>
<td>0.87</td>
<td>0.90</td>
<td>0.07</td>
<td>-7.72</td>
<td>83.58</td>
</tr>
<tr>
<td>Format Complexity</td>
<td>0.14</td>
<td>0.72</td>
<td>0.60</td>
<td>0.64</td>
<td>0.09</td>
<td>-1.05</td>
<td>5.24</td>
</tr>
<tr>
<td>Structure Complexity</td>
<td>0.15</td>
<td>0.90</td>
<td>0.61</td>
<td>0.63</td>
<td>0.17</td>
<td>-0.13</td>
<td>2.63</td>
</tr>
<tr>
<td>Data Flow Complexity</td>
<td>0.10</td>
<td>0.90</td>
<td>0.87</td>
<td>0.90</td>
<td>0.10</td>
<td>-5.64</td>
<td>39.14</td>
</tr>
<tr>
<td>Language Complexity</td>
<td>0.16</td>
<td>0.88</td>
<td>0.61</td>
<td>0.56</td>
<td>0.21</td>
<td>0.03</td>
<td>1.78</td>
</tr>
<tr>
<td>Object Point</td>
<td>42.00</td>
<td>4581.00</td>
<td>299.32</td>
<td>200.00</td>
<td>483.31</td>
<td>5.67</td>
<td>41.67</td>
</tr>
<tr>
<td>Data Point</td>
<td>29.00</td>
<td>3124.00</td>
<td>222.75</td>
<td>152.00</td>
<td>347.48</td>
<td>5.16</td>
<td>34.85</td>
</tr>
<tr>
<td>Function Point</td>
<td>6.00</td>
<td>776.00</td>
<td>53.73</td>
<td>32.00</td>
<td>94.21</td>
<td>5.36</td>
<td>35.33</td>
</tr>
<tr>
<td>Major Rule Violation</td>
<td>2.00</td>
<td>109.00</td>
<td>26.39</td>
<td>10.00</td>
<td>26.13</td>
<td>0.62</td>
<td>1.97</td>
</tr>
<tr>
<td>Medium Rule Violation</td>
<td>2.00</td>
<td>16.00</td>
<td>5.02</td>
<td>5.00</td>
<td>1.94</td>
<td>0.56</td>
<td>6.71</td>
</tr>
<tr>
<td>Minor Rule Violation</td>
<td>2.00</td>
<td>586.00</td>
<td>49.51</td>
<td>35.50</td>
<td>63.14</td>
<td>4.26</td>
<td>30.60</td>
</tr>
</tbody>
</table>

**Six interface complexity metrics** are computed between a scale of 0.0 to 1.0. A value between 0.0 and 0.4 represents low complexity and a value between 0.4 and 0.6 indicates average complexity. A value of more than 0.6 falls in the range of high complexity wherein any value above 0.8 reveals that there are significant issues with the code design [5][16].
Metrics are based on the analysis of the structure of the exchanged messages described in WSDL which becomes the basis for measuring the data complexity.
Table of Contents

1. Research Motivation and Aim
2. Related Work
3. Research Framework and Experimental Dataset
   - Research Framework
   - Experimental Dataset
4. Variables - Empirical Analysis
   - Dependent Variable
   - Predictor Variables
   - Code Metrics - Correlation Analysis
5. Feature Extraction and Selection
   - Principal Component Analysis (PCA)
   - Rough Set Analysis (RSA)
6. Extreme Learning Machines (ELMs)
7. Conclusion
   - Conclusion - I
   - Conclusion - II
8. References

Lov Kumar, Santanu Kumar Rath, Ashish Sureka
Predicting Quality of Service (QoS) Parameters using Extreme Learning Machines
Association between 37 Metrics

Lov Kumar, Santanu Kumar Rath, Ashish Sureka

Predicting Quality of Service (QoS) Parameters using Extreme Learning Machines (ELMs)
Pearson’s correlations coefficient \((r)\)

- **Black circle** represents an \(r\) value between 0.7 and 1.0 indicating a strong positive linear relationship.

- **White circle** \(r\) value between 0.3 and 0.7 indicate a weak positive linear relationship.

- **Black square** \(r\) represents a value between \(-1\) and \(-0.7\) indicating a strong negative linear relationship.

- **A white square** \(r\) represents a value between \(-0.7\) and \(-0.3\) indicating a weak negative linear relationship.

- **A blank circle** represents no linear relationships between the two variables.
# Table of Contents

1. Research Motivation and Aim
2. Related Work
3. Research Framework and Experimental Dataset
   - Research Framework
   - Experimental Dataset
4. Variables - Empirical Analysis
   - Dependent Variable
   - Predictor Variables
   - Code Metrics - Correlation Analysis
5. **Feature Extraction and Selection**
   - Principal Component Analysis (PCA)
   - Rough Set Analysis (RSA)
6. Extreme Learning Machines (ELMs)
7. Conclusion
   - Conclusion - I
   - Conclusion - II
8. References

**Lov Kumar, Santanu Kumar Rath, Ashish Sureka**

**Predicting Quality of Service (QoS) Parameters using Extreme LMs**
Identify features which are relevant in-terms of high predictive power and impact on the dependent variable and filter irrelevant features which have little or no impact on the classifier accuracy [18].
# Table of Contents

1. Research Motivation and Aim
2. Related Work
3. Research Framework and Experimental Dataset
   - Research Framework
   - Experimental Dataset
4. Variables - Empirical Analysis
   - Dependent Variable
   - Predictor Variables
   - Code Metrics - Correlation Analysis
5. **Feature Extraction and Selection**
   - Principal Component Analysis (PCA)
   - Rough Set Analysis (RSA)
6. Extreme Learning Machines (ELMs)
7. Conclusion
   - Conclusion - I
   - Conclusion - II
8. References

---

Lov Kumar, Santanu Kumar Rath, Ashish Sureka

Predicting Quality of Service (QoS) Parameters using Extreme Learning Machines (ELMs)
### Data Pre-processing Step [RSA]

<table>
<thead>
<tr>
<th>QoS</th>
<th>Selected Metrics</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Response Time</strong></td>
<td>DMR, SC, LC, WMC, Ca, LCOM3, MFA, CAM, IC, CC</td>
</tr>
<tr>
<td><strong>Availability</strong></td>
<td>FC, SC, LC, MeRV, MiRV, WMC, Ca, LCOM3, MFA, CAM, IC, CC</td>
</tr>
<tr>
<td><strong>Throughput</strong></td>
<td>ME, FC, SC, LC, MRV, MeRV, MiRV, Ce, MOA, MFA, CAM, CBM</td>
</tr>
<tr>
<td><strong>Successability</strong></td>
<td>ME, FC, SC, DFC, LC, MRV, WMC, LCOM3, LCO, DAM, MOA, CAM</td>
</tr>
<tr>
<td><strong>Reliability</strong></td>
<td>FC, SC, DFC, LC, WMC, LCOM3, LCO, MOA, MFA, CAM, CBM</td>
</tr>
<tr>
<td><strong>Compliance</strong></td>
<td>ME, FC, SC, DFC, LC, MRV, WMC, MiRV, Ca, CC, DAM, MOA, CAM, NPM</td>
</tr>
<tr>
<td><strong>Best Practices</strong></td>
<td>ME, FC, SC, DFC, LC, MRV, MiRV, WMC, Ca, NPM, MOA, MFA, CAM, CBM</td>
</tr>
<tr>
<td><strong>Latency</strong></td>
<td>DMR, ME, DC, FC, DFC, LC, MRV, NOC, NPM, LCO, MOA, CAM, IC</td>
</tr>
<tr>
<td><strong>Documentation</strong></td>
<td>ME, FC, SC, DFC, LC, MRV, MeRV, WMC, Ca, NPM, CAM, IC, CC</td>
</tr>
<tr>
<td><strong>Maintainability</strong></td>
<td>DP, Ce, LCOM3, MOA, MFA, CAM, CBM</td>
</tr>
<tr>
<td><strong>Modularity</strong></td>
<td>DMR, ME, SC, DFC, LC, DP, MRV, MiRV, WMC, Ca, MOA, IC, AMC</td>
</tr>
<tr>
<td><strong>Reusability</strong></td>
<td>MDC, DMR, FC, SC, DFC, LC, LCOM, LCOM3</td>
</tr>
<tr>
<td><strong>Testability</strong></td>
<td>ME, FC, SC, LC, MiRV, DIT, NOC, CC, RFC</td>
</tr>
<tr>
<td><strong>Interoperability</strong></td>
<td>SC, LC, MeRV, MiRV, WMC, DIT, CBO, MFA, CC</td>
</tr>
<tr>
<td><strong>Conformity</strong></td>
<td>ME, FC, DFC, LC, MRV, WMC, Ca, CAM</td>
</tr>
</tbody>
</table>

It is possible to reduce the number of features substantially and several features from the original set are found to be uncorrelated.
We investigate the performance of the ELM based classifier using three different kernel functions: linear, polynomial and RBF. Most basic, simplest and fastest is the linear kernel function which is used as a baseline for comparison with more complicated kernel functions such as polynomial and RBF. We employ four different performance metrics (MAE, MMRE, RMSE and r-value) to study the accuracy of the classifiers.
Forecasts for several predictive models are very accurate as the MAE value is less than 0.05.

The MAE value for HMS, AM, and PCA metrics for predicting Conformity is 0.03.

The predictive accuracy for response time, latency, modularity, and conformity is better than the predictive accuracy of other QoS parameters.
Observations and Results

MMRE values for ELM with RBF kernel is between 0.30 to 0.35 for response time, availability and successability and indicates good estimation ability of the classifier.

MRE values for conformity QoS parameter are as low as 0.05, 0.06, 0.10 and 0.11.

Best RMSE value in-case of ELM with linear kernel is for response time and latency QoS parameters.
The minimum RMSE value obtained is 0.08 for HMS metrics and conformity parameter in-case of linear kernel.

In-case of polynomial kernel, the performance of PCA based feature extraction technique is better for some parameters in comparison to RSA based feature selection technique.

The performance of RSA is better than PCA for some parameters. We do not observe a dominate approach between PCA and RSA.
Result of the 10-fold cross-validated paired t-test analysis. For each of the 3 kernels (Linear, Polynomial and RBF), 6 different subset of metrics are considered as input with three different performance parameters.
Research Motivation and Aim

Related Work

Research Framework and Experimental Dataset
  - Research Framework
  - Experimental Dataset

Variables - Empirical Analysis
  - Dependent Variable
  - Predictor Variables
  - Code Metrics - Correlation Analysis

Feature Extraction and Selection
  - Principal Component Analysis (PCA)
  - Rough Set Analysis (RSA)

Extreme Learning Machines (ELMs)

Conclusion
  - Conclusion - I
  - Conclusion - II

References
There exists a high correlation between Object-Oriented metrics and WSDL metrics

From t-test analysis, we infer that in most of the cases, there is the difference between the various sets of metrics in terms of the performance of the estimator is not substantial but moderate

We observe that the predictive model developed using Harry M. Sneed (HMS) metrics yields better result compared to other sets of metrics

Difference between the three kernel functions in-terms of their influence on the predictive accuracy is moderate
# Table of Contents

1. Research Motivation and Aim
2. Related Work
3. Research Framework and Experimental Dataset
   - Research Framework
   - Experimental Dataset
4. Variables - Empirical Analysis
   - Dependent Variable
   - Predictor Variables
   - Code Metrics - Correlation Analysis
5. Feature Extraction and Selection
   - Principal Component Analysis (PCA)
   - Rough Set Analysis (RSA)
6. Extreme Learning Machines (ELMs)
7. Conclusion
   - Conclusion - I
   - Conclusion - II
8. References
Final Conclusions, Summary and Takeaways

None of the feature selection technique dominate the other and one feature selection method is better than the other for some QoS parameters and vice-versa.

The polynomial kernel for ELM yields better result compared to other kernels function i.e., linear and RBF kernel functions.

The performance of the predictive model or estimator varies with the different sets of software metrics, feature selection technique and the kernel functions.

It is possible to estimate the QoS parameters using ELM and source code metrics.
Eyhab Al-Masri and Qusay H Mahmoud.  
Qos-based discovery and ranking of web services.  

Eyhab Al Masri and Qusay H Mahmoud.  
Investigating web services on the world wide web.  

D. Baski and S. Misra.  
Metrics suite for maintainability of extensible markup language web services.  
José Luis Ordiales Coscia, Marco Crasso, Cristian Mateos, Alejandro Zunino, and Sanjay Misra. 

Jos Luiis Ordiales Coscia, Marco Crasso, Cristian Mateos, and Alejandro Zunino. 
Estimating web service interface quality through conventional object-oriented metrics. 

Francisco Curbera, Matthew Duftler, Rania Khalaf, William Nagy, Nirmal Mukhi, and Sanjiva Weerawarana. 
Unraveling the web services web: an introduction to soap, wsdl, and uddi. 
References III


References IV

Barbara A Kitchenham, Lesley M Pickard, Stephen G. MacDonell, and Martin J. Shepperd.
What accuracy statistics really measure [software estimation].

Lov Kumar, Mukesh Kumar, and Santanu Ku. Rath.
Maintainability prediction of web service using support vector machine with various kernel methods.

Cristian Mateos, Marco Crasso, Alejandro Zunino, and José Luis Ordiales Coscia.

Detecting wsdl bad practices in code-first web services.
Eric Newcomer and Greg Lomow.  
*Understanding SOA with Web services.*  
Addison-Wesley, 2005.

Sunday Olusanya Olatunji, Z Rasheed, KA Sattar, AM Al-Mana, M Alshayeb, and EA El-Sebakhy.  
Extreme learning machine as maintainability prediction model for object-oriented software systems.  

H. M. Sneed.  
Measuring web service interfaces.  

D. Spinellis.  
Tool writing: a forgotten art? (software tools).  
Roman W Swiniarski and Andrzej Skowron. 
Rough set methods in feature selection and recognition. 