

Software Quality Measurement

Gruppe H

3. Januar 2018

Inhaltsverzeichnis

1	Software Quality Measurement	3
1.1	CBO – Kopplung zwischen Objektklassen	3
1.2	LCOM – Mangel an Abgeschlossenheit	4
1.3	Weitere Kenngrößen und graphische Darstellung	4
1.4	Vergleich mit einer abgeänderten Version von mallet	5

1 Software Quality Measurement

Für `mallet`¹, eine Verarbeitungssoftware für natürliche Sprache, sollen beispielhaft Kenngrößen für Softwarequalität ermittelt und ausgewertet werden. Zur Berechnung der Kenngrößen wird `ckjm`² verwendet, das alle sechs Kenngrößen nach Chidamber und Kemerer sowie zwei zusätzliche ermittelt. Die Ausgaben von `ckjm` werden mit Hilfe eines zur Verfügung gestellten, auf `matplotlib` basierenden Pythonskripts aufbereitet und in Form von Histogrammen dargestellt.

Listing 1.1: Beispiele für den Output von `mallet`

```
1 cc.mallet.extract.test.TestDocumentExtraction 7 0 0 20 31 21
   0 7
2 cc.mallet.pipe.SerialPipes$Predicate 2 1 0 2 3 1 1 2
3 cc.mallet.classify.BalancedWinnowTrainer 6 0 0 11 21 11 1 6
4 cc.mallet.util.CharSequenceLexer 20 1 0 1 45 94 14 16
5 ...
```

1.1 CBO – Kopplung zwischen Objektklassen

CBO CBO: Coupling between object classes

CBO: schlechteste Klassen	
<code>cc.mallet.optimize.OptimizerEvaluator</code>	0
<code>cc.mallet.util.tests.TestAStar\$1</code>	0
<code>cc.mallet.pipe.CharSequenceRemoveHTML\$1</code>	0

CBO: beste Klassen	
<code>cc.mallet.topics.ParallelTopicModel</code>	227
<code>cc.mallet.fst.tests.TestCRF</code>	198
<code>cc.mallet.topics.LDAHyper</code>	196

Ein hoher Wert des CBO ist erwünscht. Das bedeutet, dass Methoden mit vielen Instanzvariablen eng gekoppelt sind, was sich wiederum positiv auf die Softwarequalität auswirkt.

Ein kleiner Wert ist unerwünscht, da solche Klassen nicht mit anderen Klassen zusammen wirken.

¹Mallet af github

²ckjm auf github, Manual

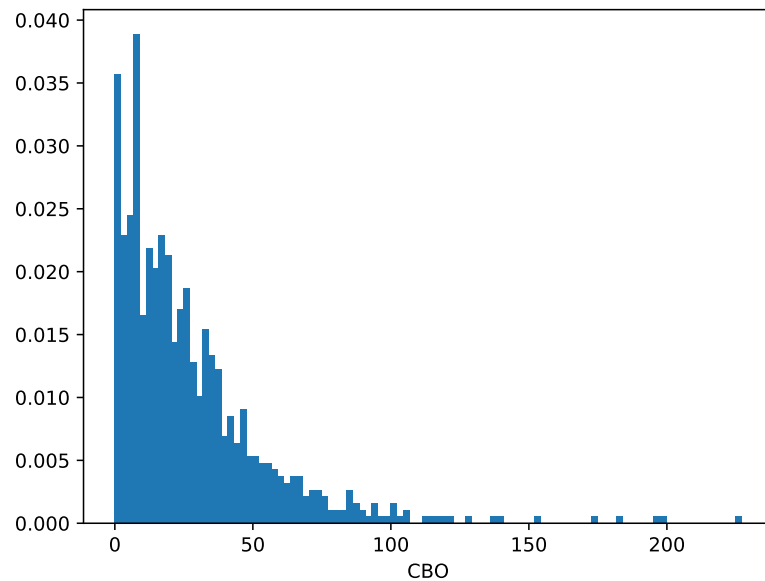


Abbildung 1.1: Coupling of Object Classes

1.2 LCOM – Mangel an Abgeschlossenheit

LCOM LCOM: Lack of cohesion in methods

LCOM: beste Klassen

<code>cc.mallet.extract.test.TestDocumentExtraction</code>	0
<code>cc.mallet.fst.LabelDistributionEvaluator</code>	0
<code>cc.mallet.optimize.OptimizerEvaluator\$ByBatchGradient</code>	0

LCOM: schlechteste Klassen

<code>cc.mallet.pipe.Pipe</code>	229
<code>cc.mallet.types.InstanceList</code>	241
<code>cc.mallet.types.Instance</code>	339

Und hier ist es dann halt andersrum.

Der LCOM-Wert drückt den Mangel an Kohäsion aus, das heißt dem Zusammenhang zwischen den Methoden. Ein großer Wert ist hier schlecht.

1.3 Weitere Kenngrößen und graphische Darstellung

Die Kenngrößen werden als normierte Histogramme dargestellt, auf der x-Achse sind Bereiche möglicher Werte der Kenngröße aufgetragen, auf der y-Achse der Anteil der

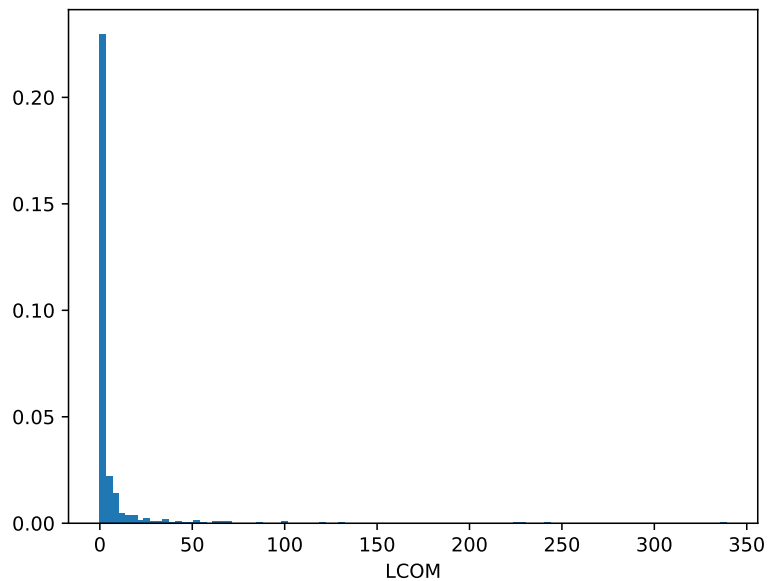


Abbildung 1.2: Lack of Cohesion

Klassen, die diesem Wertebereich zugeordnet sind.

Als weitere Kenngrößen würden von `ckjm` ermittelt:

- WMC: Weighted methods per class
- DIT: Depth of Inheritance Tree
- NOC: Number of Children
- RFC: Response for a Class
- Ca: Afferent coupling (not a C&K metric)
- NPM: Number of Public Methods for a class (not a C&K metric)

1.4 Vergleich mit einer abgeänderten Version von `mallet`

Dem Programm `mallet` werden zwei weitere Klassen `NewParallelTopicModel.java` und `TopicInferencerInterface.java` hinzugefügt. Nach dem Kompilieren wird die Analyse von `mallet` wiederholt und es werden die Kennzahlen der Klassen `NewParallelTopicModel` und `ParallelTopicModel` verglichen:

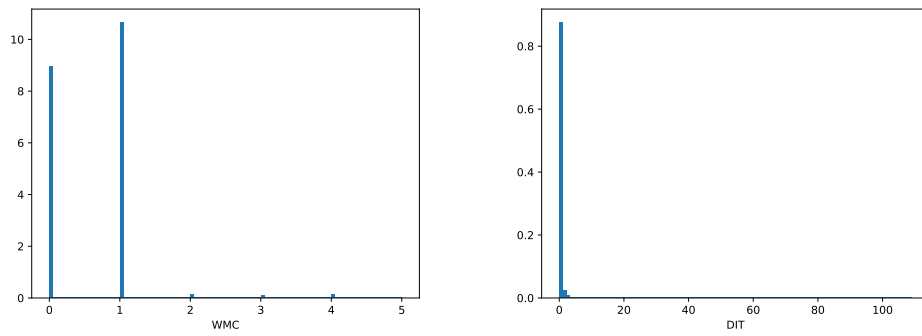


Abbildung 1.3: WMC und DIT

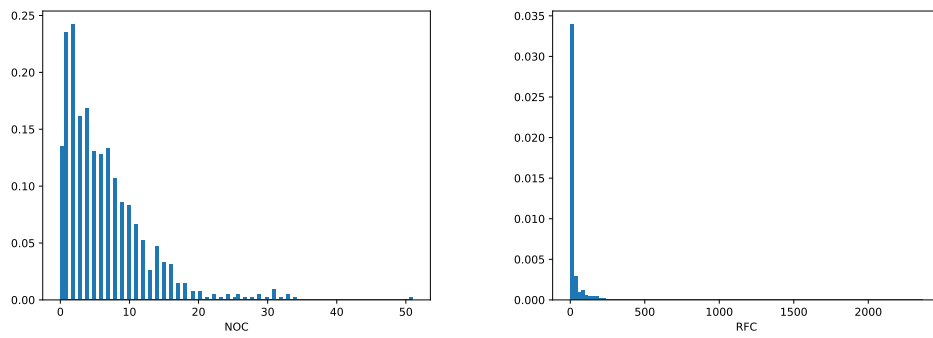


Abbildung 1.4: NOC und RFC

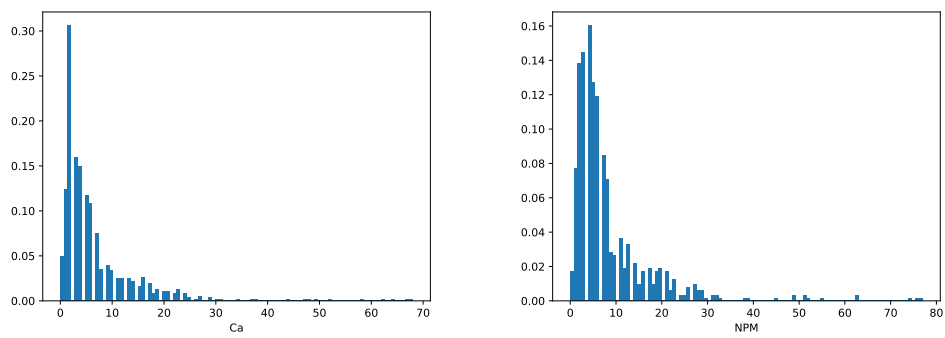


Abbildung 1.5: Ca und NPM

Klasse	WMC	DIT	NOC	CBO	RFC	LCOM	Ce	NPM
NewParallelTopicModel	68	1	0	22	233	800	0	61
ParallelTopicModel	63	1	2	21	227	627	11	58