The Conceptual Cohesion of Classes

Andrian Marcus and Denys Poshvyvanyk

Department of Computer Science
Wayne State University
Detroit, MI, USA
Motivation

- Cohesion is the degree to which the elements in a design unit (class, package) are logically related or “belong together” [Briand’oo]
- A cohesive class represents a crisp abstraction from a problem domain
- Different views of cohesion
- No accepted standard in the community
- Class cohesion can significantly affect the design, understandability, maintainability
Related Work – Class Cohesion

- Structural metrics:
  - LCOM\textsubscript{1}, LCOM\textsubscript{2} [Chidamber 94]; LCOM\textsubscript{3}, LCOM\textsubscript{4} [Hitz 94]
  - LCOM\textsubscript{5} [Henderson 96]
  - Connectivity [Hitz 94]; Coh [Briand 97, 98]
  - ICH\textsuperscript{2} [Lee 95]; TCC\textsuperscript{3}, LCC\textsuperscript{4} [Bieman 95, 98]

- Semantic metrics
  - LORM\textsuperscript{5} [Etzkorn 00]; SCDE\textsuperscript{6} [Etzkorn 02]; SCF\textsuperscript{7} [Maletic 01]

- Information entropy-based metrics; Metrics based on data mining; Slice-based metrics; etc.

1. Lack of cohesion in methods
2. Information-flow based cohesion
3. Tight class cohesion
4. Loose class cohesion
5. Logical relatedness of methods
6. Semantic class definition entropy
7. Semantic cohesion of files
Types of Cohesion

- Functional
- Informational
- Communicational
- Procedural
- Temporal
- Logical
- Coincidental
Information Retrieval Approach for Cohesion Measurement

• Using semantic information (i.e., comments, identifiers, etc.) to measure cohesion (i.e., how related are the elements of a class)

• Use information retrieval (IR) methods to extract and analyze the semantic information

• We are using Latent Semantic Indexing (LSI)
Measuring Methodology
System Representation

- Set of classes $C = \{c_1, c_2, \ldots, c_n\}$
- For each class $c \in C$, $M(c) = \{m_1, \ldots, m_k\}$
- $v_{m_k}$ and $v_{m_j}$ are the vectors corresponding to the $m_k, m_j \in M(c_i)$
- Conceptual similarity between two methods

$$CSM(m_k, m_j) = \frac{v_{m_k}^T v_{m_j}}{|v_{m_k}|_2 \times |v_{m_j}|_2}$$
The Conceptual Cohesion of Classes

- Average conceptual similarity of the methods in a class (ACSM) \( c \in C \)

\[
ACSM(c) = \frac{1}{N} \times \sum_{i=1}^{N} CSM(m_i,m_j)
\]

- Conceptual cohesion of a class (C3) \( c \in C \)

\[
C_3(c) = \begin{cases} 
ACSM(c) \text{ if } ACSM(c) > 0 \\
else & 0 
\end{cases}
\]
Example of measuring $C_3$

<table>
<thead>
<tr>
<th></th>
<th>m1</th>
<th>m2</th>
<th>m3</th>
<th>m4</th>
<th>m5</th>
</tr>
</thead>
<tbody>
<tr>
<td>m1</td>
<td>1</td>
<td>0.21</td>
<td>0.72</td>
<td>0.33</td>
<td>0.42</td>
</tr>
<tr>
<td>m2</td>
<td></td>
<td>1</td>
<td>0.28</td>
<td>0.91</td>
<td>0.66</td>
</tr>
<tr>
<td>m3</td>
<td></td>
<td></td>
<td>1</td>
<td>0.37</td>
<td>0.27</td>
</tr>
<tr>
<td>m4</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>0.89</td>
</tr>
<tr>
<td>m5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>

Conceptual similarities between the methods in class c.
$\text{ACSM}(c)=0.5 \rightarrow C_3(c) = 0.5$
Shortcomings of C3

• Are two classes with the same C3 value equally cohesive? (SD of the CSM values)
• Measure the influence of highly related methods in a class with a low C3 cohesion
• Define a new measure based on the counting mechanism utilized in LCOM2
  – Do not take into account intersections of methods based on common attribute usage
  – Count intersections of method pairs based on the CSM value between them and the ACSM
Lack of Conceptual Similarity between Methods (LCSM)

- Let $M_i = \{m_j \mid (m_i, m_j) \in E, m_i \neq m_j\}$ be the set of neighbor methods of $m_i$ (with which $m_i$ has a higher CSM value than the average).
- Let $P = \{(M_i, M_j) \mid M_i \cap M_j = \emptyset\}$
- Let $Q = \{(M_i, M_j) \mid M_i \cap M_j \neq \emptyset\}$
- Lack of conceptual similarity is

$$
\text{LCSM}(c) = \begin{cases} 
|P| - |Q| & \text{if } |P| > |Q| \\
0 & \text{else}
\end{cases}
$$
Example of Measuring LCSM

\[
\begin{array}{ccccc}
& m1 & m2 & m3 & m4 & m5 \\
m1 & 1 & 0.21 & \textbf{0.72} & 0.33 & 0.42 \\
m2 & & 1 & 0.28 & \textbf{0.91} & \textbf{0.66} \\
m3 & & & 1 & 0.37 & 0.27 \\
m4 & & & & 1 & \textbf{0.89} \\
m5 & & & & & 1
\end{array}
\]

ACSM(c) = 0.5
M1 = \{m3\},
M2 = \{m4, m5\},
M3 = \{m1\},
M4 = \{m2, m5\},
M5 = \{m2, m4\}

\[
\begin{array}{ccccc}
& M1 & M2 & M3 & M4 & M5 \\
M1 & & \emptyset & \emptyset & \emptyset & \emptyset \\
M2 & & & \emptyset & m5 & m4 \\
M3 & & & & \emptyset & \emptyset \\
M4 & & & & & m4 \\
M5 & & & & & \emptyset
\end{array}
\]

|P| = 7; |Q| = 3;
LCSM(c) = 7 - 3 = 4
Limitations

- C3 and LCSM do not take into account polymorphism and inheritance
- Method invocation, parameters, attribute references, and types are of interest only at identifier level
- C3 and LCSM do not make distinction between constructors, accessors, and other method stereotypes. Some of these methods can artificially increase or decrease cohesion
C₃’ and LCSM’

- Same measurement as C₃ and LCSM
- Eliminate comments from the analysis
- Keep identifiers only
- Assess the influence of comments quality over C₃ and LCSM
Case Study

- Compare C₃, C₃’, LCSM, and LCSM’ with [LCOM₁-LCOM₅], Coh, C, ICH, TCC, and LCC

- WinMerge with 51KLOC and 11K comments
- Metrics computed for 34 classes with 522 methods
- Structural metrics computed with Columbus [Ferenc’04], C₃ and LCSM – our tool

- Analysis of correlations between metrics
Results

- C3 and C3’ very close values (WinMerge has 20% of code as comments)
- LCSM and LCSM’ are less conclusive in this respect, but the differences are still not major
- C3 and LCSM do not correlate – interesting!
- Significant correlations between C3 and ICH, and C3 and LCOM5 – not major surprise
- No significant correlation between any structural metric and LCSM – somewhat surprising! – expected LSOM2 to correlate
Interesting Cases

• “It is after all possible to have a class with high internal, syntactic cohesion but little semantic cohesion” – Henderson-Sellers

<table>
<thead>
<tr>
<th>Class name</th>
<th>C3</th>
<th>LCOM2</th>
</tr>
</thead>
<tbody>
<tr>
<td>IVSSItem</td>
<td>0.64</td>
<td>528</td>
</tr>
<tr>
<td>IVSSDatabase</td>
<td>0.635</td>
<td>136</td>
</tr>
<tr>
<td>IVSSItemOld</td>
<td>0.632</td>
<td>465</td>
</tr>
<tr>
<td>BCMenuData</td>
<td>0.434</td>
<td>0</td>
</tr>
<tr>
<td>CDirDoc</td>
<td>0.294</td>
<td>0</td>
</tr>
<tr>
<td>RescanSuppress</td>
<td>0.392</td>
<td>1</td>
</tr>
</tbody>
</table>
Interesting Cases

- IVSSItem, IVSSDatabase, IVSSItemOld – wrapper classes with few or no data members. Wrappers tend to group together methods that are conceptually similar or have similar usage.

- BCMenuData is a class that implements a “property container”. Small class with many accessor methods. Many unique identifiers. Cohesive – C3 did not capture it.

- RescanSuppress only three methods: constr., destr., clear – C3 limited in this case.

- On the top and bottom values, C3 and LCOM2 agreed.
Metrics are Complementary

- Structural metrics tell us if a class is built cohesively
- Semantic/conceptual metrics tell us if a class is written cohesively
- We desire both -> increase maintainability
Future Work

• More case studies

• Investigate new measuring mechanism using the semantic information (e.g. like LCOM₃)

• Combine with structural metrics