Extracting and Modeling Product Line Functional Requirements

Nan Niu, Steve Easterbrook

Department of Computer Science
University of Toronto, Canada
Email: {nn, sme}@cs.toronto.edu

Sept. 11, 2008
Product Lines Are Everywhere

Big Mac

Filet-O-Fish

McRib

Big N' Tasty
Core Assets: Benefit or Burden?

⇒ Reuse is planned, enabled, and enforced
  ⇒ Core assets: design for reuse
  ⇒ Products: design with reuse

⇒ Core assets ≠ actual product
  ⇒ Not worth building by itself

⇒ Domain analysis
  ⇒ Heavyweight & no rules
Extractive SPL Adoption Model

⇒ [Krueger, 2001]

⇒ Reusing existing products for the SPL’s initial baseline

⇒ SMEs (small and medium-sized enterprises)

⇒ Rather under-specify than over-specify

⇒ Reactive development can overcome under-specifying
Outline

- Problem
- Extracting
- Modeling
- Conclusions
Requirements Assets

⇒ Principles [Pohl et al., 2005]
  ⇒ Focus more on external variability (visible to customers) and less on internal variability (useful to implementers)
  ⇒ Focus more on what varies (variation point) and less on how it varies (variants)

⇒ Challenges of extraction
  ⇒ Where, what, & how to extract?
  ⇒ How to represent (model) the results?
Illustrative Example

3.1 Functions
...
3.1.3.2 Marking
AMS shall store students’ assignments.
Instructor shall create the marking rubric.
Marker shall mark students’ assignments.
AMS shall apply late policy automatically.
AMS can generate report for each section.
...
3.1.3.4 Remark
Students can request remarking to markers.
AMS shall access marked assignments.
...

(a) SRS for AMS (Assignment Marking System)

3.1 Functional requirements
...
3.1.1 The EMS shall time-stamp any information sent online.
...
3.1.4 A professor must create a marking scheme for an assignment. A professor specifies mark breakdown, and records this information using a marking rubric.
...
3.1.8 A TA shall mark any portion of an assignment.
...

(b) SRS for EMS (Electronic Marking System)

⇒ Where to extract?
⇒ Natural language documents
What to Extract?

- **Functional requirements**
  - Salient features directly observable by the users and other stakeholders
  - Basis for aligning and optimizing quality requirements

- **Research question**
  - Given a natural language document, how can its characterizing attributes, which relate to system functionalities, be produced?
Single-term Indices

⇦ Verb_Freq [John, 2001]
⇦ Highest frequencies of occurrence

⇦ Verb_INFO
⇦ INFO(w) = $-\log_2(P\{w\})$

<table>
<thead>
<tr>
<th>Verbs</th>
<th>Freq.</th>
<th>Verbs</th>
<th>INFO</th>
</tr>
</thead>
<tbody>
<tr>
<td>mark</td>
<td>64</td>
<td>accept</td>
<td>11.67</td>
</tr>
<tr>
<td>access</td>
<td>18</td>
<td>install</td>
<td>11.67</td>
</tr>
<tr>
<td>submit</td>
<td>14</td>
<td>highlight</td>
<td>11.67</td>
</tr>
<tr>
<td>use</td>
<td>9</td>
<td>update</td>
<td>10.67</td>
</tr>
<tr>
<td>notify</td>
<td>9</td>
<td>check</td>
<td>10.09</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Multi-term Indices

[Maarek et al., 1991]

- Using LAs (lexical affinities) to profile software libraries
- Information Retrieval based technique: cost, scalability, and domain transportability
- Shortcoming #1: strictly two-word long
  - Domain-aware: “marking rubric” → “marking_rubric”
- Shortcoming #2: mixed modifier-modified relations
  - NL program analysis: “Verb-Direct Object” pairs can represent action-oriented concerns [Fry et al., 2008]
FRPs

- Functional Requirements Profiles
  - Action-oriented concerns that bear a high information value of a document
  - Model user-visible system functionalities
  - Represented by “verb-direct object” pairs

- Sample auto-marker FRPs
  - mark assignment, view grade, create marking rubric...
Profiling An Auto-Marker SRS

<table>
<thead>
<tr>
<th>Verbs</th>
<th>Freq.</th>
<th>Verbs</th>
<th>INFO</th>
<th>FRPs</th>
<th>$\rho$</th>
</tr>
</thead>
<tbody>
<tr>
<td>mark</td>
<td>64</td>
<td>accept</td>
<td>11.67</td>
<td>mark assignment</td>
<td>106.75</td>
</tr>
<tr>
<td>access</td>
<td>18</td>
<td>install</td>
<td>11.67</td>
<td>access assignment</td>
<td>96.31</td>
</tr>
<tr>
<td>submit</td>
<td>14</td>
<td>highlight</td>
<td>11.67</td>
<td>divide number</td>
<td>69.85</td>
</tr>
<tr>
<td>release</td>
<td>14</td>
<td>define</td>
<td>11.67</td>
<td>notify instructor</td>
<td>61.97</td>
</tr>
<tr>
<td>provide</td>
<td>12</td>
<td>calculate</td>
<td>10.67</td>
<td>modify information</td>
<td>54.68</td>
</tr>
<tr>
<td>use</td>
<td>9</td>
<td>update</td>
<td>10.67</td>
<td>release assignment</td>
<td>44.58</td>
</tr>
<tr>
<td>notify</td>
<td>9</td>
<td>check</td>
<td>10.09</td>
<td>change password</td>
<td>43.10</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

**Quantity of information:**

$$\rho((\{u_1, u_2\}, f)) = \frac{f \times \text{INFO}(\{u_1, u_2\})}{\log_2(\prod_{u_1} P(u_1) \times \prod_{u_2} P(u_2))}$$
Effort

⇒ Linear
⇒ Empirical observation: 98% of lexical affinities relate words which are separated by at most 5 words within a single sentence
⇒ 5-word window: a property of English

⇒ OpenNLP
⇒ Stemming & POS tagging
⇒ Overhead is low
Effectiveness

Precision (accuracy) = \( \frac{C}{B} \)
Recall (coverage) = \( \frac{C}{A} \)
Modeling FRPs

Previous work

-[Pohl et al., 2005]
  - OVM (orthogonal variability model): variation points, variants, dependencies, and constraints
  - Consistent variability management across requirements, design, realization, and testing

-[Liaskos et al., 2006]
  - Using Fillmore’s semantic cases to identify variability in goal models
  - Goal concern (e.g., “send message”) $\approx$ FRP
Uncovering Variation Structures

⇒ Essential semantic cases (variation dimensions)
  ⇒ Agentive
  ⇒ Objective
  ⇒ Process
  ⇒ Conditional

(a) Cases for FRPs

(b) Orthogonal variability model
Semantic Case Analysis

- Focus more on WHAT and less on HOW
  - Reactively enriched

- Treating each FRP as a variation point
  - Every product in the product line should address the FRPs (domain’s action themes)
  - Other concerns can be variation points

- Manual effort is indispensable
  - NLP tools can help
Discovering Variability Dependencies

\( \Rightarrow \) Intra-FRP variability: mandatory or optional
  \( \Rightarrow \) Heuristic 1: If a case is associated with only one value, then the case has one mandatory variant.
  \( \Rightarrow \) Heuristic 2: (in the paper).

\( \Rightarrow \) Inter-FRP variability: requires or excludes
  \( \Rightarrow \) Heuristic 3: If FRP\(_2\) is conditional to FRP\(_1\), then there exists a vp\_requires\_vp constraint from FRP\(_1\) to FRP\(_2\).
  \( \Rightarrow \) Heuristic 4: (in the paper).
Outline

- Problem
- Extracting
- Modeling
- Conclusions

costly
Aligning Quality Requirements

[Niu & Easterbrook, RE 2006]

Tackling terminological interferences with the Repertory Grid Technique

FRPs: common ground
Cluster Analysis

[Niu & Easterbrook, SPLC 2008]

User: identify, browse, prioritize features

Designer: system decomposition and modularization
Concluding Remarks

⇒ FRPs
costly no more
Acknowledgments

⇒ CSC340 (Winter 2007) students and staff at University of Toronto

⇒ Jennifer Campbell, Jia Wang, Sotirios Liaskos

⇒ Anonymous RE’08 reviewers