Second-Milestone of term project, Software Engineering (CS2310)

Zichuan (Jerry) Ye

Project Title:

"A conceptual study of Product & Service Customization System on mobile platform".

Second milestone accomplishment:

Precise mathematical definitions of specification is needed when implementing I-card/C-card concepts in Slow Intelligence Systems. In general, transitions in SIS run through a pathway that starts from a specific problem and finishes when a solution occurs:

'problem' -> 'enumerator' -> 'super-component' -> 'eliminator' -> 'concentrator' -> 'solution'

In this case, a product/service customization system (PSCS) can then be specified given mathematical representations that describes components and operations from initial problem to final solution. The initial problem in a PSCS is to provide a best service which matches the most number of needs from a customer. A solution therefore is the best service from available resources. Mathematical representations of problem and solution are as following:

Problem: Product service that optimally satisfies a customer's need under given resources.

Enumerator: enumerates the possible features that are expected to be preferred by potential customers. It generates a set of attributes when each tuple represents a specific product.

$expected\_feature = \{ef_1, ef_2, \dots, ef_i\};$

Super-component: processes the needs from customers. It contains a variety of attributes that **partially** agree with attributes from enumerator. As a result, multiple transitions are possible in matching them.

$customer\_need = \{cn_1, cn_2, \dots, cn_j\};$

Tester: a tester is need for multiple decision circles such that timing can be controlled.

Eliminator: transitions from enumerator to super-component can therefore generate a maximum of $n = i*j$ attributes, as shown in eliminator.

$product\_candidate = \{pc_1, pc_2, \dots, pc_m \mid m < n\};$

Concentrator: the resulting attributes in *product_candidate* are then further concentrated in a concentrator. A concentrated selection algorithm is applied here to merge similar attributes accommodating customers' need as according to available resources.

$concentrated\_result = \{cr_1, cr_2, \dots, cr_x\};$

Solution: service that contains the largest number of matching attributes between products and customer needs.

$solution = \{s_1, s_2, \dots, s_y\}.$