

# **Design and Evaluation of a Method to Reduce the Lexical Ambiguity of Requirement Specifications**

Stephen Boyd (B.Eng(Comp.Sys))

A Thesis Submitted for the Degree of

**DOCTOR OF PHILOSOPHY**

University of Technology, Sydney  
Faculty of Engineering and Information Technology

2009

# Certificate of Authorship/Originality

---

I certify that the work in this thesis has not previously been submitted for a degree nor has it been submitted as part of requirements for a degree, except as fully acknowledged within the text.

I also certify that the thesis has been written by me. Any help that I have received in my research work and the preparation of the thesis itself has been acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

Signature of Student

Production Note:

Signature removed prior to publication.

---

# Acknowledgement

---

The author would like to thank three people for their support throughout this research:

1. I would like to thank my primary supervisor Prof. Didar Zowghi from the University of Technology, Sydney for her regular face-to-face discussions, problem solving advice, literature recommendations, and detailed review comments throughout the years of my research.
2. I would like to thank my secondary supervisor A/Prof. Vincenzo Gervasi from the University of Pisa, Italy for his in-depth review and accurate comments on my design solution and evaluation method.
3. Most importantly, I would like to thank my wife Alia for her understanding, patience, guidance, and continual support.

# Preface

---

The author has previously published sections of this thesis in conference proceedings:

- Boyd, S., Zowghi, D., and Farroukh, A., 2005, 'Measuring the Expressiveness of a Constrained Natural Language: An Empirical Study', *Proceedings of the 13th International Conference on Requirements Engineering (RE'05)*, Paris, France, pp.339-349.
- Boyd, S., Zowghi, D., and Gervasi, V., 2007, 'Optimal-Constraint Lexicons for Requirement Specifications', *International Working Conference on Requirements Engineering Foundations for Software Quality (REFSQ '07)*, Trondheim, Norway, pp.203-217.

The author has been industry-supervisor for four engineering Capstone Project Reports that relate to this PhD thesis:

- Adi-wijaya, A., 2003, 'BOYDA: Object Oriented Requirements Engineering', University of Technology, Sydney, Unpublished Capstone Project Report.
- Farroukh, A., 2005, 'To define and use expressiveness to empirically derive the verbs of BOYDA for a particular domain', Faculty of Engineering, University of Technology, Sydney, Unpublished Capstone Project Report.
- Saeed, M., 2005, 'To define an unambiguous, disciplined and repeatable process for the coupling of BOYDA with an existing system modelling language', University of Technology, Sydney, Unpublished Capstone Project Report.
- Selvarajan, R., 2005, 'To develop a software tool that implements the BOYDA language and BOYDA modelling rules', University of Technology, Sydney, Unpublished Capstone Project Report.

# Table of Contents

---

Chapter 1	Introduction.....	2
1.1.	Research Problem .....	2
1.2.	Research Question .....	4
1.3.	Motivation to Answer Research Question.....	4
1.4.	Research Scope .....	5
1.5.	Research Contributions .....	6
1.6.	Roadmap .....	8
Chapter 2	Literature Review.....	10
2.1.	Purpose.....	10
2.2.	Systems Engineering.....	10
2.3.	Requirements Engineering .....	12
2.3.1.	What is a Requirement.....	12
2.3.2.	Requirements Engineering Process .....	13
2.3.3.	Properties of a High-Quality Requirements Specification .....	13
2.3.4.	Important Quality Factors.....	14
2.4.	Requirements Ambiguity.....	15
2.4.1.	Linguistic Ambiguity .....	17
2.4.1.1.	Lexical ambiguity .....	18
2.4.1.2.	Syntactic ambiguity .....	19
2.4.1.3.	Semantic ambiguity.....	20
2.4.1.4.	Pragmatic ambiguity .....	21
2.4.2.	Why are Requirements Ambiguous? .....	22
2.4.3.	Reducing Ambiguity in Requirements .....	23
2.4.3.1.	Learn to write less ambiguously and less imprecisely.....	23
2.4.3.2.	Learn to detect ambiguity and imprecision .....	24
2.4.3.3.	Use a restricted natural language.....	25
2.5.	Controlled Natural Language .....	26
2.5.1.	Evolution of Controlled Natural Languages .....	27
2.5.1.1.	Development of Controlled Natural Language .....	30
2.5.1.2.	Claimed Quality Improvements.....	31
2.6.	Quality Model .....	33
2.6.1.	Readability.....	34
2.6.2.	Conventionality.....	36
2.6.3.	Writability.....	36
2.6.4.	Expressiveness .....	37
2.6.5.	Correctness.....	38
2.7.	Summary and Research Opportunities.....	39

Chapter 3	Research Question and Methodology .....	41
3.1.	Purpose.....	41
3.2.	Research Question .....	41
3.2.1.	Explanation of Research Question .....	41
3.2.2.	Relationship to Contributions .....	42
3.3.	Research Methodology.....	43
3.3.1.	Research Outputs .....	45
3.3.1.1.	Constructs.....	45
3.3.1.2.	Models .....	45
3.3.1.3.	Methods .....	45
3.3.1.4.	Instantiations.....	46
3.3.2.	Research Activities.....	46
3.3.2.1.	Build .....	46
3.3.2.2.	Evaluate .....	46
3.3.2.3.	Theorize .....	46
3.3.2.4.	Justify.....	47
3.4.	Summary.....	47
Chapter 4	Solution .....	48
4.1.	Purpose.....	48
4.2.	Constructs.....	48
4.2.1.	Notation .....	48
4.2.2.	Universe .....	49
4.2.3.	Constants .....	49
4.2.4.	Variables .....	49
4.2.5.	Functions.....	49
4.2.6.	Predicates .....	49
4.3.	Model .....	50
4.3.1.	Similar( $W_r, W_o$ ).....	50
4.3.2.	Correct( $W_r, \text{Context}(W_o)$ ) .....	50
4.3.3.	Conventional( $W_r, \text{Context}(W_o)$ ) .....	51
4.3.4.	Unambiguous( $W_r, \text{Context}(W_o)$ ).....	51
4.4.	Theories .....	52
4.4.1.	Ideal( $W_r, \text{Context}(W_o)$ ) .....	52
4.4.2.	Optimal( $W_r, \text{Context}(W_o)$ ).....	53
4.5.	Method .....	53
4.5.1.	Meaning( $W$ ).....	54
4.5.1.1.	Background.....	54
4.5.1.2.	Method .....	56
4.5.1.3.	Implementation .....	56

4.5.2.	Senses(W)	59
4.5.2.1.	Background	59
4.5.2.2.	Method	59
4.5.2.3.	Implementation	59
4.5.3.	Frequency(W)	60
4.5.3.1.	Background	60
4.5.3.2.	Method	60
4.5.3.3.	Implementation	60
4.5.4.	Width(W)	60
4.5.4.1.	Background	60
4.5.4.2.	Method	61
4.5.4.3.	Implementation	61
4.5.5.	Similarity( $W_1, W_2$ )	61
4.5.5.1.	Background	62
4.5.5.2.	Method	62
4.5.5.3.	Implementation	63
4.5.6.	Replaceability( $W_1, W_2$ )	64
4.5.6.1.	Background	65
4.5.6.2.	Method	68
4.5.6.3.	Implementation	68
4.5.7.	Sequence of Activities	68
Chapter 5	Evaluation	71
5.1.	Purpose	71
5.2.	Evaluation Model	71
5.2.1.	Metrics	72
5.2.1.1.	Makes_sense( $P, W_r, \text{context}(W_o)$ )	72
5.2.1.2.	Would_use( $P, W_r, \text{context}(W_o)$ )	73
5.2.2.	Variables	74
5.2.2.1.	Independent Variable(s)	74
5.2.2.2.	Dependent Variable(s)	74
5.3.	Evaluation Method	75
5.3.1.	Confounding Effects	77
5.3.1.1.	Multiple Parts of Speech	77
5.3.1.2.	Time	78
5.3.2.	Constants	78
5.3.3.	Hypotheses	78
5.3.4.	The Data	79
5.3.5.	Data Collection	80
5.3.6.	Questionnaire Design	80

5.3.6.1.	Questions.....	80
5.3.7.	Sample Size.....	85
5.3.8.	Hypothesis Testing.....	85
5.3.8.1.	Z-Test for Two-Proportions.....	87
5.3.9.	Participants.....	88
5.3.10.	Reliability.....	89
5.3.10.1.	Interrater Reliability.....	89
5.3.10.2.	Test-retest Reliability.....	89
5.3.11.	Validity.....	89
5.3.11.1.	Internal Validity.....	89
5.3.11.2.	External Validity.....	91
5.3.12.	Evaluation Limitations.....	94
5.4.	Evaluation Results.....	95
5.4.1.	Specification ( $S_0$ ).....	95
5.4.2.	Results of Manual Word Sense Disambiguation.....	96
5.4.2.1.	Validation of Manual Word Sense Disambiguation.....	97
5.4.3.	Execution of the Software Prototype.....	99
5.4.4.	The Sample.....	101
5.4.5.	Questionnaire Results.....	109
5.4.5.1.	Respondents.....	109
5.4.5.2.	Data Collected.....	110
5.4.6.	Hypothesis Testing.....	112
5.4.6.1.	Z-Test Criteria.....	112
5.4.6.2.	Hypothesis Test Results.....	113
5.4.6.3.	Summary of Hypothesis Test Results.....	114
5.4.7.	Reliability.....	114
Chapter 6	Conclusion.....	116
6.1.	Research Summary.....	116
6.2.	Answer to the Research Question.....	117
6.2.1.	Limitations of Answer.....	117
6.3.	Contributions Revisited.....	117
6.4.	Implications for Research.....	118
6.4.1.	Dynamic Constraint.....	119
6.4.2.	Optimal Constraint.....	119
6.4.3.	Empirical Evidence.....	120
6.5.	Implications for Industry.....	120
6.6.	Suggestions for Future Research.....	121
6.6.1.	Automatic Word Sense Disambiguation.....	121
6.6.2.	Integration with an existing RM Tool.....	121

# Abstract

---

The requirements engineering process has been criticised for its immaturity. Firstly, in the context of safety-critical systems, missing, misunderstood, and erroneous requirements have been attributed as the cause of many safety-system faults; and secondly, in the context of project success factors, many IT projects have identified requirement defects as a primary cause of being over-time or over-budget. Ambiguity is a requirement defect that is commonly associated with challenged IT projects, however there are but few empirical studies on how ambiguity can be reduced or eliminated from requirement specifications.

Eliminating the ambiguity inherent within a requirement specification is the seemingly unattainable ambition of the systems engineering zealot. This is because ambiguity is considered an unavoidable side-effect of using natural language, and most requirement specifications are written in natural language. One proposed solution to the ambiguity problem is to express requirements in Controlled Natural Language (CNL). CNLs enforce grammatical and/or lexical constraints to reduce the inherent ambiguity of natural language without sacrificing correctness, readability, or expressiveness. There is, however, a view in the literature that CNLs are overly restrictive and unnatural to read and write. Furthermore, the design and development of CNLs is both labour-intensive and time-intensive.

This thesis describes how a requirements specification can be automatically re-expressed in a way that significantly reduces its lexical ambiguity, without significantly reducing its correctness or conventionality. The thesis specifically focuses on lexical ambiguity, since this is the form of ambiguity most attributable to the lexicon used to express the specification. The term re-expression is used to distinguish this approach from that of CNLs, since the lexicon is not static, but is optimally selected on a word-by-word basis such that lexical ambiguity is minimised, whilst correctness and conventionality are maximised. Fundamental to the optimal word selection is a new concept: replaceability( $W_1, W_2$ ), which is the degree to which word  $W_1$  can replace word  $W_2$ . The replaceability equation developed within this thesis is a function of semantic similarity, polysemy, frequency, and lexical width.

We implement a software prototype, and execute it on an existing industry-specification. A controlled experiment is used to measure the effects of the re-expression in terms of correctness, conventionality, and lexical ambiguity. Data are collected from project stakeholders using a questionnaire-style approach, and hypothesis testing is used to decide whether or not the optimal re-expression has significantly reduced lexical ambiguity without significantly reducing correctness or conventionality.