

**A Computational Model for Task-adapted Knowledge
Organisation: Improving Learning through Concept Maps
Extracted from Lecture Slides**

A thesis
submitted in fulfilment
of the requirements for the Degree
of
Doctor of Philosophy
by

Atapattu Mudiyanseelage Thushari Dilhani Atapattu

Supervised by

Associate Professor Katrina Falkner
Dr Nickolas Falkner
Dr Edward Palmer

School of Computer Science
University of Adelaide
November 2014

Declaration

I certify that this work contains no material which has been accepted for the award of any other degree or diploma in any university or other tertiary institute and, to the best of my knowledge and belief, contains no material previously published or written by another person, except where due reference has been made in the text. In addition, I certify that no part of this work will, in future, be used in a submission for any other degree or diploma in any university or other tertiary institution without the prior approval of the University of Adelaide and where applicable, any partner institution responsible for the joint-award of this degree.

I give my consent to this copy of my thesis, when deposited in the University Library, being made available for loan and photocopying, subject to the provisions of the Copyright Act 1968.

I also give permission for the digital version of my thesis to be made available on the web, via the University's digital research repository, the Library catalogue and also through web search engines, unless permission has been granted by the University to restrict access for a period of time.

.....

Atapattu Mudiyansele Thushari Dilhani Atapattu

This thesis is dedicated to my beloved husband

Abstract

This thesis presents a framework for automatically generating concept maps from lecture slides. A concept map is recognised as a valuable educational visualisation tool, which assists students in organising, sharing and representing knowledge. Expert maps (also known as expert concept maps) are prepared by domain experts with the intention to serve as scaffolding to facilitate learning. Automated concept map generation provides an alternative solution to the labour-intensive and time-consuming process of manually constructing *expert maps*. Therefore, the main objective of this thesis is to develop techniques to extract maps from lecture slides, ensuring that auto-generated concept maps may be utilised as a positive alternative to expert maps. This process is known as concept map mining (CMM).

The particular interest of this thesis is on CMM from *lecture slides*, due to their wide usage within the teaching context and the poor support of sequentially-structured lecture slides in aiding learners in identifying relationships between information. In general, semantically and syntactically missing and ambiguous text in lecture slides make it undesirable for adopting previously developed algorithms for CMM.

Within this thesis, a set of Natural Language Processing (NLP) algorithms are developed to support concept-relation-concept triple extraction to form concept maps. To support knowledge extraction and to overcome the noise associated with text, this work utilises contextual features specific to lecture slides. The natural layout of the lecture slides is incorporated to organise the extracted triples in a hierarchy. Structural (e.g. co-occurrence, term frequency) and graph-based features (e.g. degree of centrality) are utilised to rank the triples according to their importance within the domain. A series of evaluation studies in this thesis identify promising results, with several case studies demonstrating a strong positive correlation between auto-generated concept maps and human generated maps. These results indicate that this research provides an effective and efficient alternative to expert maps.

Auto-generated concept maps can be utilised to provide scaffolding in the problem solving context, in particular supporting students who are lacking the required skills. Even though this application has been studied previously, these studies do not specifically focus on the relevance of information to learning. To fill this gap, this thesis investigates an approach to provide more relevant concept maps to a given problem. In pursuit of this goal, a framework capable of automatically extracting concept maps according to the given problems (named task-adapted concept maps) is developed, utilising auto-generated concept maps from lecture slides as domain knowledge. In order to investigate the effect of task-adapted concept maps as scaffolding for learning, an evaluation study was undertaken, with students in the task-adapted

concept map scaffolding group demonstrated statistically significant learning gain compared to the students who received lecture slides or full concept maps as scaffolding.

Acknowledgements

I would like to express my heartfelt gratitude to my supervisors Associate Professor Katrina Falkner, Dr Nickolas Falkner and Dr Edward Palmer for the immense support, guidance, valuable advice, encouragement and constructive feedback given throughout the entire time period to complete this research.

I would also like to thank the members of the Computer Science Education Research Group (CSER) for their generous support in making this research project a great success. I am grateful to Dr Amali Weerasinghe for assisting me with in-class experiments and Dr Rebecca Vivian for assisting me to obtain ethics approval for the research. I would also like to thank Mr. Hamid Tarmazadi for assisting me with developing the web-based prototype. I am also grateful to all the HDR colleagues and teaching assistants who helped me with the experiments and all the participants involved in the experiments.

Finally, I take this opportunity to express my sincere gratitude to my beloved parents and husband for their encouragement and support given during my studies.

This research is supported by a Divisional Scholarship of University of Adelaide and Research Training Scheme of Australian Government.

Table of Contents

List of Tables	iv
List of Figures	vi
List of Publications	ix
Chapter 1 Introduction	1
1.1 Concept Map Extraction from Lecture Slides	3
1.2 Task-adapted Concept Maps as Scaffolding	7
1.3 Research Objectives and Questions	8
1.4 Guide to the Thesis.....	9
Chapter 2 Concept Mapping: A Review	11
2.1 Overview of Concept Mapping	11
2.2 Theoretical Basis of Concept Mapping	11
2.3 Applications of Concept Mapping	13
2.4 Knowledge Organisation versus Text Representations.....	17
2.5 Concept Maps as Scaffolding.....	21
2.6 Summary	28
Chapter 3 Concept Map Mining: A Review	29
3.1 Overview of Concept Map Mining	29
3.2 Concept Map Mining Systems and Methods	30
3.3 Evaluation of Concept Map Mining.....	42
3.4 Concept Map Mining Applications in the Educational Context	45
3.5 Summary	49
Chapter 4 Concept Map Mining from Lecture Slides	50
4.1 Design of the Concept Map Mining Framework	51
4.1.1 Structured Data Analysis: Design	54
4.1.2 Knowledge Acquisition: Design	58
4.1.3 Ranking: Design.....	61
4.2 Development of the Concept Map Mining Framework	64
4.2.1 Development Environment	64
4.2.2 Natural Language Annotation	64

4.2.3	Structured Data Analysis	67
4.2.4	Knowledge Acquisition	74
4.2.5	Ranking	86
4.2.6	Concept Map Visualisation	89
4.2.7	Summary	93
Chapter 5 Task-adapted Knowledge Organisation.....		95
5.1	Design of the Task-Adapted Scaffolding Framework.....	98
5.2	Development of the Task-Adapted Scaffolding Framework.....	106
5.2.1	Development Environment.....	106
5.2.2	Natural Language Annotation	106
5.2.3	Question Processor	107
5.2.4	Domain model	110
5.2.5	Task-adapted Concept Map Extractor.....	112
5.2.6	Visualisation.....	113
5.3	Design of the Web-based Prototype	114
5.4	Development of the Web-based Prototype.....	115
5.4.1	Development Environment.....	115
5.4.2	Interaction Design	115
5.4.3	Database Design	117
5.4.4	Web Interfaces.....	117
5.4.5	Quantitative Data.....	119
5.5	Summary	119
Chapter 6 Evaluation.....		121
6.1	Evaluation of the Concept Map Mining Framework.....	123
6.1.1	Evaluation Measures	123
6.1.2	Subject Domain	124
6.1.3	Evaluation of Concept Extraction	124
6.1.4	Evaluation of Noise Detection.....	128
6.1.5	Evaluation of Pronoun Resolution.....	130
6.1.6	Evaluation of Triple Extraction	132

6.1.7	Evaluation of Concept Map Extraction.....	136
6.1.8	Summary	143
6.2	Evaluation of the Task-Adapted Scaffolding Framework.....	144
6.2.1	Experimental Design.....	144
6.2.2	Data Analysis	150
6.2.3	Results and Discussion.....	151
6.2.4	Summary	166
Chapter 7	Conclusion.....	167
7.1	Main Contributions	167
7.2	Future Directions.....	170
7.3	Concluding Remarks.....	172
Appendix A	173
Appendix B	177
Appendix C	180
Appendix D	195
References	197

List of Tables

Table 2.1: Comparison between concept mapping and PowerPoint (Kinchin, et al., 2008)	20
Table 2. 2: Types of instructional scaffolding (Alibali, 2006)	22
Table 2.3: Bloom Taxonomy of educational objectives (Bloom, et al., 1956).....	26
Table 2.4: Question classification by (Graesser, et al., 1992)	27
Table 3.1: Comparison of concept map mining systems.....	40
Table 3.2: Knowledge organisation from lecture slides	41
Table 4.1: Issues associated with the corpus	53
Table 4.2: Performance of pronoun resolution by Leskovec et al. (2004)	56
Table 4.3: Examples of demonstrative determiners	57
Table 4.4: Part-of-speech tags (word-level) used in Penn Treebank project.....	65
Table 4.5: NLP Annotations of an example sentence	66
Table 4.6: Features of co-occurrence calculation for noise detection in slide-level	68
Table 4.7: Features for pronoun resolution	69
Table 4.8: Phrase- and clause-level Penn TreeBank tags.....	72
Table 4.9: Commonly used link types in the Link Grammar parser	79
Table 4.10: Features for triple extraction	81
Table 4.11: Regular expression patterns to identify concepts	82
Table 4.12: Hierarchy levels assigned for slide data.....	84
Table 4.13: Concept importance determined by location.....	86
Table 4.14: Concept importance determined by grammatical structure.....	87
Table 4.15: Elements and attributes of CXL	90
Table 5.1: Comparison between the components of typical QA system and TASF	100
Table 5.2: Question types defined in QA systems (Dali, et al., 2009; Kolomiyets & Moens, 2011; Olney, et al., 2012).....	100
Table 5.3: Question types supported by TASF	102
Table 5.4: NLP annotations of an example question.....	106
Table 5.5: Penn Treebank tags for question texts.....	107
Table 5. 6: Important features for task-adapted concept map extraction	112
Table 6.1: Contingency table.....	123

Table 6.2: Agreement between two evaluators	123
Table 6.3: Evaluation results of concept extraction	126
Table 6.4: Comparison between F-measure and inter-rater agreement.....	127
Table 6.5: Evaluation results of noise detection	129
Table 6.6: Statistics of pronouns discovered in the corpus	131
Table 6.7: Evaluation results of pronoun resolution	131
Table 6.8: Comparison between F-measure and inter-rater agreement.....	134
Table 6.9: Spearman’s ranking correlation (r_s) of baseline model.....	138
Table 6.10: Spearman’s ranking correlation (r_s) of linguistic feature model	139
Table 6.11: Best fit parameter values for structural features	140
Table 6.12: Spearman’s ranking correlation (r_s) of structural feature model.....	140
Table 6.13: Visual annotations used by observers	149
Table 6.14: Statistics of student groups	151
Table 6.15: Students’ prior experience on knowledge organisations techniques.....	151
Table 6.16: Descriptive statistics of pre-test scores	152
Table 6.17: Descriptive statistics of learning gain	153
Table 6.18: Test of homogeneity of variances (learning gain).....	154
Table 6.19: Tests of normality	154
Table 6.20: Summary results of ANOVA.....	155
Table 6.21: Summary of Tukey HSD post-hoc test	156
Table 6.22: Pearson correlation between the time spent on scaffolding and learning gain	159
Table 6.23: Pros and cons about concept maps as scaffolding	165
Table 6.24: Suggestions for improvements of the system	165
Table D.1: What do you think about concept maps used in this study to answer questions? ...	195
Table D.2: What are the issues related to concept maps in this study?.....	195
Table D.3: Do you think a tool to automatically generate concept maps from lecture slides will help answering questions?.....	195
Table D.4: Which form of resources do you prefer if our tool made available through CS Forums in future?.....	195
Table D.5: Would you like a tool which can extract partial concept maps to assist answering questions?.....	196
Table D.6: What type of courses do you think this kind of tool will be more useful?.....	196

List of Figures

Figure 1.1: An example concept map.....	1
Figure 1.2: A sample slide set	6
Figure 2.1: Distinction between rote and meaningful learning (Novak & Canas, 2006)	12
Figure 2.2: Concept map structures a) spoke b) chain c) net (Kinchin, et al., 2000)	14
Figure 2.3: Overview of transformation cycle (Kinchin, et al., 2008)	18
Figure 2.4: An example of a transformation cycle (Bradley, et al., 2006; Kinchin, et al., 2008)	19
Figure 2.5: Illustrative model of scaffolding.....	22
Figure 2.6: Concept mapping tasks model (Ruiz-Primo, 2004).....	25
Figure 3.1: Concept map mining process (Villalon & Calvo, 2008).....	29
Figure 3.2: An example for generalisation/specialisation (is-a) relation.....	33
Figure 3.3: Concept map activity of Guru (Person, et al., 2012).....	45
Figure 4.1: High-level architecture of the CMMF	53
Figure 4.2: Sample slide to illustrate incompleteness of sentences.....	55
Figure 4.3: Sample slide to illustrate semantically ambiguous sentences	55
Figure 4.4: Contextual features embedded in presentation framework.....	58
Figure 4.5: (a) Sample lecture slide (b) corresponding concept map generated using CMMF ...	60
Figure 4.6: Term frequency using ‘Operating System’ slide	62
Figure 4.7: Degree of co-occurrence using ‘Operating System’ slide	62
Figure 4.8: An illustration of degree centrality	63
Figure 4.9: Typographic information in a sample lecture slide.....	63
Figure 4.10: Parse tree of an example sentence	66
Figure 4.11: The structure of the Apache POI for Microsoft PowerPoint documents	67
Figure 4.12: An example slide with demonstrative determiners	71
Figure 4.13: A parse tree of an example NP ‘Generic software process models include’	72
Figure 4.14: Parse tree of an example VP ‘should start with well-understood requirements’	72
Figure 4.15: Sample lecture slide from Software Engineering course	73
Figure 4.16: Decision tree of CMMF	75
Figure 4.17: Overview of SVO extraction from simple sentences.....	75
Figure 4.18: Parse tree for an example sentence with highlighted SVO	76

Figure 4.19: Triple extraction from an example sentence using (a) baseline algorithm (Rusu, et al., 2007) (b) CMMF.....	77
Figure 4.20: Parser tree of an example complex sentence	78
Figure 4.21: Linkage diagram of an example complex sentence	79
Figure 4.22: Concepts occur in multiple levels of the slides	82
Figure 4.23: An example of hierarchy extraction (with ‘relation label’).....	84
Figure 4.24: An example of hierarchy extraction (without ‘relationship label’)	84
Figure 4.25: Sample CXL file.....	90
Figure 4.26: An example CourseCMap	92
Figure 5.1: Example question 1 (a) text-based answer (b) task-adapted concept map	95
Figure 5.2: Example question 2 (a) text-based answer (b) task-adapted concept map	96
Figure 5.3: High-level architecture of the task-adapted scaffolding framework.....	99
Figure 5.4: (a) CourseCMap (b) TSKCMap	105
Figure 5.5: Parser tree of an example question	107
Figure 5.6: A skeleton tree used to describe the algorithm 5.1 (Dali et al., 2009).....	109
Figure 5.7: parser tree and corresponding triples for (a) question 1 (b) question 2	109
Figure 5.8: An example of concept map merging	111
Figure 5.9: Interaction flow diagram	115
Figure 5.10: Database design for the web-based prototype	117
Figure 5.11: Screen shots of (a) user login (b) quiz (c) scaffolding resource	118
Figure 5.12: Final overview of the research.....	120
Figure 6.1: Design of the evaluation studies	121
Figure 6.2: Distance between human and computer ranks against percentage of concepts ($r_s = 0.813$)	141
Figure 6.3: Experimental design	145
Figure 6.4: Histogram of pre-test scores	153
Figure 6.5: Normal Q-Q plot of learning gain in each group.....	155
Figure 6.6: Students’ opinion on having concept maps or lecture slides as scaffolding to answer questions.....	160
Figure 6.7: Students’ issues related to concept maps.....	161
Figure 6.8: Students’ opinion on CMMF	162
Figure 6.9: Students’ preferred form of scaffolding	162

Figure 6.10: Students' opinion on task-adapted concept maps as scaffolding to answer questions	163
Figure 6.11: Most suitable Computer Science courses for concept mapping	164
Figure C.1: Task-adapted concept map as scaffolding for question 1.....	185
Figure C.2: Concept map as scaffolding for question 1	186
Figure C.3: Concept map with highlighted problem solving context as scaffolding for question 1	187
Figure C.4: Task-adapted concept map as scaffolding for question 2.....	188
Figure C.5: Concept map as scaffolding for question 2	189
Figure C.6: Concept map with highlighted problem solving context as scaffolding for question 2	190

List of Publications

Atapattu, T., Falkner, K. and Falkner, N. 2012. Automated Extraction of Semantic Concepts from Semi-Structured Data: Supporting Computer-Based Education through the Analysis of Lecture Notes. In Liddle, S., Schewe, K. D., Tjoa, A. and Zhou, X (Eds.), *Database and Expert Systems Applications (DEXA 2012)*. Vol. 7446, pp. 161-175. Springer Berlin Heidelberg.

Atapattu, T., Falkner, K. and Falkner, N. 2014. Acquisition of Triples of Knowledge from Lecture Notes: A Natural Language Processing Approach. In Stamper, J., Pardos, Z., Mavrikis, M. and McLaren, B. M. (Eds). *Proceedings of the 7th International Conference on Educational Data Mining (EDM 2014)*. London, United Kingdom. pp. 193-196.

Atapattu, T., Falkner, K. and Falkner, N. 2014. Evaluation of Concept Importance in Concept Maps Mined from Lecture Notes – Computer Vs Human. *Proceedings of the 6th International Conference on Computer Supported Education (CSEDU 2014)*. Barcelona, Spain. pp. 75-84.

Atapattu, T., Falkner, K. and Falkner, N. 2015. Educational Question Answering Motivated by Question-Specific Concept Maps. *Proceedings of the 17th International Conference on Artificial Intelligence in Education (AIED 2015)*. Madrid, Spain.

Atapattu, T., Falkner, K. and Falkner, N. 2015. Task-adapted Concept Map Scaffolding to Support Quizzes in an Online Environment. *Proceedings of the 20th Annual Conference on Innovation and Technology in Computer Science Education (ITiCSE 2015)*. Vilnius, Lithuania.

Chapter 1

Introduction

Concept mapping is recognised as a valuable educational visualisation technique, which assists students in organising, sharing and representing knowledge (Novak & Gowin, 1984). A concept map is a diagram consisting of concepts, usually represented by circles or boxes, connected by directed edges to form relationships, which are represented by annotated lines between the concepts. Figure 1.1 illustrates an example of a concept map that describes ‘*what are birds?*’ (Canas & Novak, 2009a). A concept can be a noun or a short noun phrase, and the relationship label is usually presented as a verb or verb phrase. According to the example in Figure 1.1, ‘*birds*’, ‘*feathers*’, and ‘*eggs*’ represent concepts and connecting words such as ‘*lay*’, ‘*have*’ represent relationship labels. The concept-relation-concept triple forms a proposition, which represents a meaningful statement to interpret. In the example concept map, *birds-lay-eggs* and *feathers-help to-fly* triples form propositions. Concept maps represent an overview of domain knowledge, employing a hierarchical organisation scheme with the most general concept at the top, and the more specific concepts arranged below (Novak & Canas, 2006). Cross-links between concepts in different segments of the map allow learners to understand the interconnections between different domains, facilitating creative thinking (Novak & Gowin, 1984). For instance, ‘*high metabolism provides energy*’ represents a cross-link that creates interconnections between the concepts ‘*rapid digestive systems*’ and ‘*food*’.

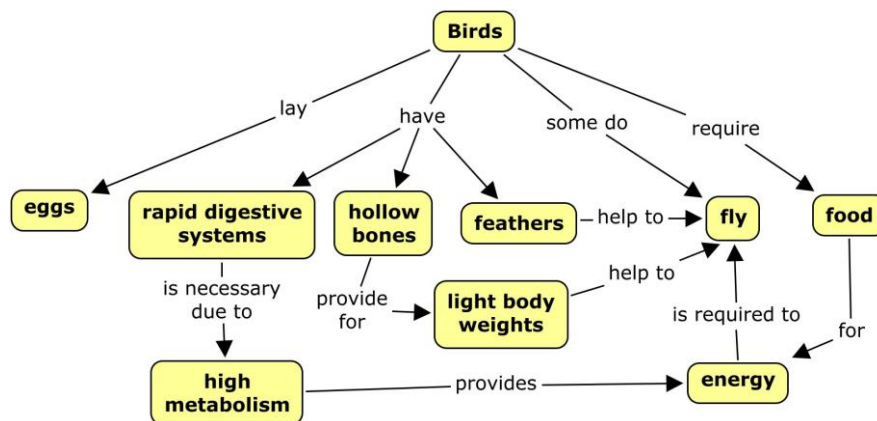


Figure 1.1: An example concept map

The use of concept maps as a way of supporting learning is well established. There has been shown to be a significant increase in meaningful learning when using concept maps, and further gains in the use of concept maps to assist in the assessment of learning (Coffey et al., 2003;

Heinze-Fry & Novak, 1990; Irvine, 1995; Nesbit & Adesope, 2006; Novak, 2010; Novak & Canas, 2006). Meaningful learning refers to the integration of relevant prior knowledge with new information (Ausubel, 2000). However, the widespread adoption of concept mapping as a technique is hindered by the substantial assistance and feedback required by learners constructing concept maps (Canas & Novak, 2009b; Edmondson, 2005; Ferry, Hedberg, & Harper, 1997; Fisher et al., 1990; Novak & Canas, 2006). To overcome the issues associated with manual concept map construction, the idea of *expert maps* is introduced (Coffey et al., 2003; Novak & Canas, 2006). Expert maps are prepared by domain experts (e.g. teachers) with the intention to utilise the expert maps for learning, comparison, evaluation and scaffolding purposes (Chang, Sung, & Chen, 2001; Ruiz-Primo, 2004; Ruiz-Primo & Shavelson, 1996). Scaffolding is the support given during the learning process with the intention of helping the student achieves his or her learning goals. However, the construction of expert maps places an additional workload on the academics involved, requiring multifaceted expertise of the domain, and the creation of complex concept maps.

Therefore, this thesis investigates an automated approach to produce concept maps as good as expert maps. Recent efforts within the field of research, work toward automated approaches to extract concept maps from text, with the aim of utilising auto-generated concept maps as expert maps for scaffolding purposes (Alves, Pereira, & Cardoso, 2001; N.-S. Chen, Kinshuk, Wei, & Chen, 2008; Clariana & Koul, 2004; Lau, Chung, Song, & Huang, 2008; Olney, 2010; Valerio & Leake, 2006, 2012; Villalon & Calvo, 2009, 2011; Wang, Cheung, Lee, & Kwok, 2008; Zouaq & Nkambou, 2008).

1.1 Concept Map Extraction from Lecture Slides

The main objective of this thesis is to develop techniques to extract concept maps from *lecture slides*, enabling the use of auto-generated concept maps as a positive alternative to expert maps. The motivation for concept map extraction from lecture slides arises from three key supporting reasons;

1. Concept maps are a highly effective educational tool, grounded in important learning theories (Ausubel, 1968; Novak, 1993; Novak & Canas, 2006; Novak & Gowin, 1984).
2. The need to reduce the issues associated with manual construction of concept maps by learners or construction of expert maps (Canas & Novak, 2009b; Chang et al., 2001; Edmondson, 2005; Fisher et al., 1990).
3. Sequentially-structured lecture slides are not supportive of effective knowledge organisation (Brandt et al., 2001; Kinchin, Chadha, & Kokotailo, 2008; Kinchin, Hay, & Adams, 2000).

The notion of concept mapping was first introduced in 1972, grounded in the Assimilation theory of Ausubel which states “*learning takes place by the integration of new concepts and propositions into existing concept and propositional framework held by the learner*” (Ausubel, 1968; Novak & Gowin, 1984). The hierarchical nature of concept maps supports this learning theory by identifying the general concepts held by the learner prior to introducing more specific concepts. The integration of relevant prior knowledge in learning new information is known as *meaningful learning*, enabling the construction of effective knowledge structures. Meaningful learning is the opposite of *rote learning*, where, the learner simply memorises facts with little or no effort to relate new knowledge with relevant existing information in learner’s cognitive structures (Novak & Canas, 2006). In addition, the use of concept maps to identify relations between concepts and more progressively, identification of cross-links between concepts, involves a high level of cognitive performance (Bloom, Englehart, Furst, Hill, & Krathwohl, 1956; Edmondson, 2005). The ability of concept maps to facilitate meaningful learning through well-known learning theories and its support for high levels of cognitive performance distinguishes concept maps from other popular knowledge representations such as mind maps, semantic networks, knowledge maps, and cognitive maps (Coffey et al., 2003).

Since 1997, more than 500 empirical studies have utilised concept maps for various learning and teaching activities or assessments (McClure, Sonak, & Suen, 1999; Nesbit & Adesope, 2006) including the adoption of concept mapping for assessing understanding (Edmondson, 2005), identifying both valid and invalid ideas held by learners (Novak, 2002) and the use in

assessing the learners' conceptual change (Markham, Mintzes, & Jones, 1994; Pearsall, Skipper, & Mintzes, 1997).

Although a meta-analysis of 55 studies showed that the process of a student constructing a concept map has a higher learning gain (measured using effect size of d ; $d = 0.82$) than a student studying a concept map ($d = 0.37$) (Nesbit & Adesope, 2006), the use of concept mapping as a learning technique is not popular for several reasons, particularly within tertiary education (Kinchin, 2001; P. A. Martin, 2008; Ruiz-Primo & Shavelson, 1996).

The manual construction of concept maps requires substantial assistance and feedback from teachers, particularly for novice and rote learners, and even after proper training in appropriate processes (Chang et al., 2001). Rote learners are generally poor at constructing effective knowledge structures, particularly if they spend their learning process memorising facts in contrast to meaningful learning (Cliburn, 1990; Hsu, 2004). Most learners struggle to identify correct concepts, relations and their hierarchical organisation. More specifically, learners struggle to identify relation labels and cross-links between concepts both of which involve higher mental load (Canas & Novak, 2009b; Edmondson, 2005; Ferry et al., 1997; Fisher et al., 1990; Novak & Canas, 2006). This occurs mainly when traditional teaching and learning resources such as lecture materials and text books focus more on isolated pieces of information than their inter-relationships (Taber, 1994). Other works have made similar statements, emphasising the lack of human awareness of knowledge organisation techniques, and the general preference for sequentially-structured textual representations such as writing using natural language sentences over creating network models leads to lack in popularity of concept mapping (P. A. Martin, 2008). All, Huycke & Fisher (2003) stated that "*helping students to formalise concept links is one of the most challenging tasks in teaching*". In a qualitative study with 48 grade seven Biology students, 94% students expressed that concept map construction is an effort-demanding task and requires scaffolding and feedback during map construction (Chang et al., 2001).

In order to overcome the issues associated with manual concept map construction, more recent works introduce the idea of *expert maps* (also known as expert concept maps) (Chang et al., 2001; Hauser, Nuckles, & Renkl, 2006; Novak & Canas, 2006; Ruiz-Primo, Schultz, Li, & Shavelson, 2001; Valerio & Leake, 2012). Within the context of concept mapping, expert maps are identified as concept maps which are usually constructed by human experts within a particular domain or topic. Expert maps are introduced using different terms in different studies such as *expert skeleton maps* (Novak & Canas, 2006), *teacher-prepared maps* (Horton et al., 1993), *pre-constructed concept maps* (Nesbit & Adesope, 2006), *worked-out map* (Hauser et al., 2006) and *expert-generated maps* (Olney, Cade, & Williams, 2011; Villalon & Calvo, 2008).

The main purpose of expert map is to serve as a scaffolding to facilitate learning, while minimising the requirement for prior experience in concept map construction. This support can be categorised differently based on the learning activity.

Firstly, Novak & Canas (2006) introduced the idea of *expert skeleton maps* which provided a learner with a small concept map prepared by an expert in the domain. Typically this map would contain 6-10 concepts with proper linking words and a good hierarchy. The learner was required to extend this concept map, typically in the order of 3-4 times the number of concepts held in an expert skeleton map.

Secondly, Chang et al. (2001) and Ruiz-Primo et al. (2001) introduced the idea of the ‘fill-in-the-structure’ approach (Naveh-Benjamin, Lin, & McKeachie, 1995) which was grounded in the ‘completion strategy’ theory (Naveh-Benjamin et al., 1995; Pass, 1992). This approach provided learners with an incomplete framework of an expert map with some nodes and links displayed as blank. Expert maps were provided as scaffolding for learning and problem solving in the works of Bagno & Eylon (1997), Horton et al. (1993), Hauser et al. (2006) and Valerio & Leake (2012). Even though the purpose of introducing expert maps as scaffolding is to overcome the issues associated with manual construction of concept maps by learners, constructing expert maps is still a labour-intensive and time-consuming process.

Thus, recent efforts in the area of concept mapping work toward semi- or fully automated approaches to extract concept maps from various text sources such as text books (Olney, 2010), text documents (Alves et al., 2001), news stories (Y.-H. Tseng, Chang, Rundgren, & Rundgren, 2010), academic articles (N.-S. Chen et al., 2008), and student essays (Villalon & Calvo, 2009) with the aim of providing useful educational tools with minimal human intervention. This process is known as concept map mining (CMM) (Villalon & Calvo, 2008).

The particular interest of this thesis is on concept map extraction from *lecture slides*. Lecture slides in this thesis context are used to introduce the general knowledge source, which represent the documents that are used to deliver a particular lecture in classroom or online environment. The terms ‘slide’ and ‘slide set’ will be used in the thesis when referring to a single slide and a collection of slides respectively which are built on top of a presentation framework. Figure 1.2 illustrates a sample slide set obtained from *Operating System* course (Silberchatz, Galvin, & Gagne, 2012). This thesis demonstrates the concept map extraction from lecture slides using Microsoft PowerPoint lecture slides as the basis material, as it is the most commonly used lecture slides format (Apperson, Laws, & Scepanzky, 2008; Atkins-Sayre, Hopkins, Mohundro, & Sayre, 1998; Bartsch & Cobern, 2003; Frey & Birnbaum, 2002; Sugahara & Boland, 2006).

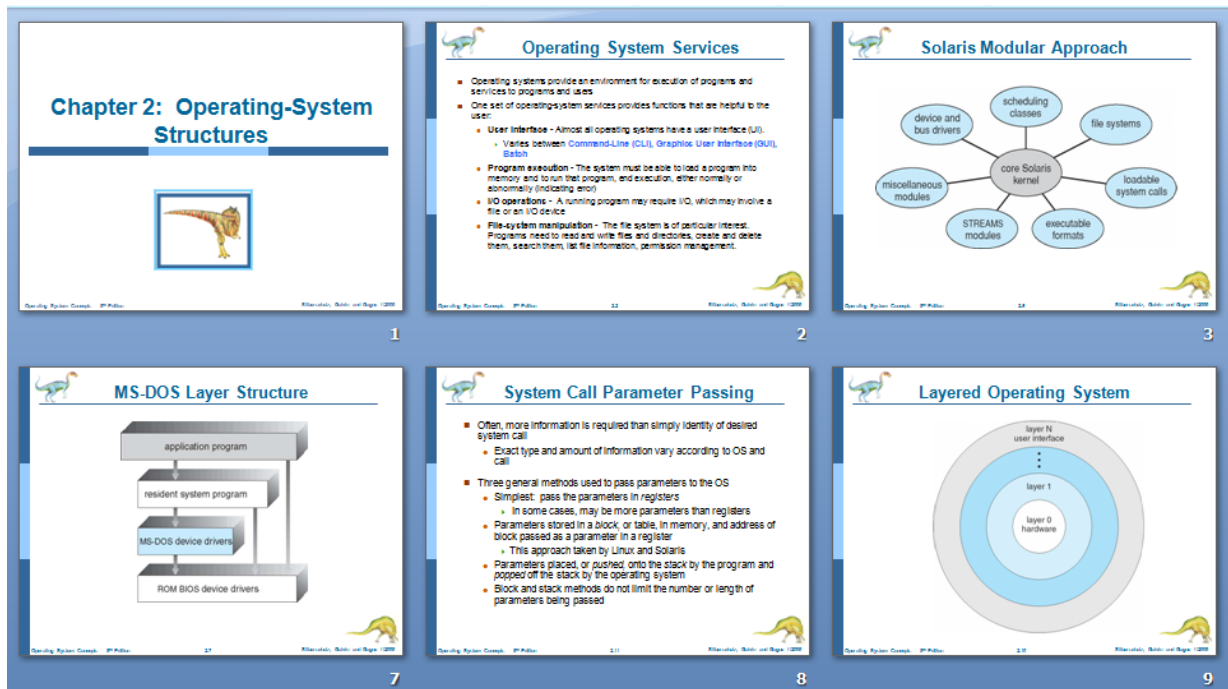


Figure 1.2: A sample slide set

Despite the richness of the knowledge structures of the lecturers, the ultimate form of the lecture slides is a series of bullet points (Kinchin, 2006a). The sequentially-structured lecture slides is not very supportive in aiding learners in identifying complex relationships between information, making it difficult to integrate new knowledge with existing knowledge, which can result in poor knowledge organisation (Brandt et al., 2001). Choosing lecture slides as the knowledge source for concept map generation has potential benefits as teachers have already invested substantial time and effort in producing legible slides and their use provides a reliable start for knowledge acquisition of the domain.

This research can be utilised across any discipline with availability of sufficiently mineable lecture slides. This thesis demonstrates the concept map extraction using the Computer Science domain. The ability of concept mapping to promote meaningful learning within the Computer Science discipline is discussed in the recent works of Calvo et al. (2011), Larraza-Mendiluze & Garay-Vitoria (2013) and Sanders et al. (2008).

In pursuit of the goal of developing techniques to extract concept maps from lecture slides of equivalent quality to expert maps, the main hypothesis is constructed as *‘It is possible to develop auto-generated concept maps of lecture slides to strongly correlate with human constructed maps’*.

1.2 Task-adapted Concept Maps as Scaffolding

Auto-generated concept maps from various text sources are utilised to design various pedagogical activities including creating concept map activities in Intelligent Tutoring Systems (ITS) such as *Betty's Brain* (Leelawong & Biswas, 2008) and *Guru* (Person, Olney, D'Mello, & Lehman, 2012), and question generation (Olney, C., & K., 2012; Olney et al., 2011). In contrast to the creation of pedagogical activities using concept maps mined from educational materials, concept map mining algorithms can be applied to students' data or tasks to extract concept maps to visualise students' written essays (Villalon & Calvo, 2009, 2011), posts in the discussion forums (Lau et al., 2008), and answers to exam questions (S.-M. Chen & Bai, 2010; Lee, Lee, & Leu, 2009).

Concept maps as well as elements of concept maps, act as scaffolding resources to support organising knowledge. For instance, elements of concept maps such as lists of concepts, relation labels, and hints are provided to students to fill incomplete maps (Chang et al., 2001; Ruiz-Primo et al., 2001). Expert maps, or auto-generated concept maps which can act as expert maps, are utilised as scaffolding to improve learning outcomes (Hauser et al., 2006; Horton et al., 1993). More importantly, expert maps have a strong positive effect on problem solving success (Bagno & Eylon, 1997; Valerio & Leake, 2012), which is the particular interest in this thesis. Okebukola (1992) demonstrated that *concept mappers* were significantly more successful in solving problems.

Even though problem solving using concept maps as scaffolding has been studied previously, existing research does not specifically focus on the relevance of information to learning. To fill this gap, this thesis investigates an approach to provide the most relevant information to answer specific *questions*, using concept maps generated from lecture slides. More precisely, the concept maps utilised as scaffolding in this thesis will be adapted to the given problem. This process is known as 'task-adapted knowledge organisation'. Questions in this context are similar to formative or summative questions provided through online learning environments to engage and motivate students, focus attention, guide learning, provide opportunity for practice and self-assessment (Dillon, 1988; Hunkins, 1972; Wilen, 1986).

Answering questions using task-adapted concept maps as scaffolding thus stands as the second goal of the thesis. The interest in investigating the effect of task-adapted concept maps as scaffolding for answering questions is supported by two key statements. First, according to an early study by Eylon & Reif (1984) which stated "*higher levels of the hierarchy should preferentially contain information most important for the domain of tasks*", it was demonstrated that performance in problem solving tasks was significantly better when hierarchies were

adapted to the given tasks. In addition, according to Canas & Novak (2006), concept maps were more effective when they were produced to answer a question (known as a *focus question*) rather than using them to represent general knowledge in a domain or topic. The creation of a concept map to answer a question is similar to that proposed in this thesis. However, the proposed approach will support more effective scaffolding, due to its task-adapted nature. In addition, providing more relevant information as scaffolding reduces the risk of information overload which therefore lessens the anxiety, stress, alienation and learning disorientation among learners (Dias & Sousa, 1997; Edmunds & Morris, 2000).

In order to investigate the effect of task-adapted concept maps as scaffolding for answering questions, the second hypothesis is constructed as '*students who receive task-adapted concept map as scaffolding will have an increased learning gain compared to those who did not receive task-adapted concept maps*'.

1.3 Research Objectives and Questions

To support the main objective of this thesis which develops techniques to extract concept maps from lecture slides, the research is divided into stages, designed to address several research questions.

- 1) *Can computer-generated concepts be used as an alternative to human extracted concepts?*
- 2) *Can computer-generated concept-relation-concept triples be used as an alternative to human extracted triples?*
- 3) *Can computer-generated concept maps be used as an alternative to human extracted concept maps?*
- 4) *Can computer-based replacements of pronouns be used as an alternative to human identified replacements of pronouns?*

To support the second motivation of the thesis which investigates the effect of task-adapted concept maps as scaffolding for answering questions, this research expects to conduct an evaluation study to compare the task-adapted concept map condition with other scaffolding techniques including lecture slides, full concept maps, and full concept maps with highlighted problem solving context. In addition, this research addresses the following research questions;

- 1) *Is there any significant correlation between the time spent on scaffolding and the learning gain?*
- 2) *What are the students' opinions of concept mapping and scaffolding?*

1.4 Guide to the Thesis

Chapter 2: Concept Mapping: A Review

Chapter 2 provides a detailed review of the background of concept mapping. The chapter starts with a discussion on learning theories that are grounded with concept mapping and discusses empirical studies which adopted concept maps in classroom education. Then, the benefits of adopting knowledge organisation techniques over text representations are discussed. Finally, previous studies which utilise concept maps as scaffolding are summarised.

Chapter 3: Concept Map Mining: A Review

Chapter 3 provides a detailed review related to concept map mining from various text sources, their evaluation methodologies and demonstrates how this thesis resides amongst related literature. After that, application areas with the use of concept maps mined from text is discussed. Finally, previous studies which utilise concept maps as scaffolding for problem solving are summarised.

Chapter 4: Concept Map Mining from Lecture Slides

Chapter 4 presents the design and development of the concept map mining framework from lecture slides. The framework is presented according to a definition of concept map mining by Villalon & Calvo (2008) including the extraction of concept, concept-relation-concept triple, hierarchy, and the ranking. Additionally, an overview of noise detection, elimination of syntactic- and semantic-level ambiguities from lecture slides is presented.

Chapter 5: Task-Adapted Knowledge Organisation

Chapter 5 presents the design and development of the framework for answering questions with task-adapted concept maps as scaffolding. The first half of the chapter provides a detailed overview of task-adapted concept map extraction process. Afterwards, the design and development of a web-based prototype is presented.

Chapter 6: Evaluation

The evaluation chapter is presented as two stages. Firstly, the chapter presents an algorithm evaluation to measure the effectiveness of concept map mining framework. This chapter includes three main studies to evaluate concept and triple extraction algorithms, and ranking. Another two supplementary studies evaluate pronoun resolution and noise detection to improve the performance of the concept map mining framework. Secondly, the chapter presents an

evaluation study to measure the effectiveness of task-adapted scaffolding framework. Finally, a qualitative study is presented to analyse students' feedback.

Chapter 7: Conclusion

The conclusion chapter discusses how this thesis has fulfilled its objectives as specified here and the conclusions drawn based on the hypotheses. The main contributions to the discipline are summarised followed by a description of each technical contributions. Finally, the chapter concludes with a description of limitations, which suggesting avenues for future research and closing remarks.

Chapter 2

Concept Mapping: A Review

This chapter includes an overview of concept mapping, the theoretical basis and the application areas of concept mapping within the educational context. The chapter discusses the benefits of utilising knowledge organisation techniques over sequentially-structured text representations. Further, this chapter includes the theoretical basis and the adoption of concept maps for scaffolding purposes to facilitate learning. The chapter concludes with the gaps identified in the field of research and provides ways to address them.

2.1 Overview of Concept Mapping

Concept maps are an effective educational tool that can assist learners in organising, sharing and representing knowledge (Novak & Gowin, 1984). A concept map is a diagram consisting of concepts, usually represented in circles or boxes, connected by directed edges to form relationships. A *concept* is defined as a perceived regularity in events or objects, designated by a label. The label can be a word or symbol. Words on the connecting line between concepts specify the relationship between the two concepts (Novak & Gowin, 1984). The concept-relation-concept triple (or triplet) forms a proposition, which is a meaningful statement to interpret. A concept map represents an overview of domain knowledge, employing a hierarchical organisation scheme with the most general concept at the top, and the more specific concepts arranged below (Novak & Canas, 2006). Cross-links between concepts in different segments of the map allow learners to understand the interconnections between different domains, facilitating creative thinking (Novak & Gowin, 1984). Concept maps are typically constructed to understand a particular event, *focus* question or domain through the organisation of knowledge (Novak & Canas, 2006).

2.2 Theoretical Basis of Concept Mapping

The notion of concept mapping was first introduced in 1972 based on the learning psychology of Ausubel (Ausubel, 1968; Novak, 1990; Novak & Gowin, 1984), as a solution to overcome the difficulty in identifying changes in knowledge using interview techniques (Edmondson, 2005). Early works on identifying changes in children's understanding of science concepts involved interviewing children (Novak, 1990). The cognitive learning theory of Ausubel (known as *assimilation theory*) states "*learning takes place by the integration of new concepts and propositions into existing concept and propositional framework held by the learner*"

(Ausubel, 1968). The existing knowledge structure held by learners is also referred as the individual's *cognitive structure*. The development of 'concept' originates when children in the ages of birth to three years recognise regularities around them and begin to identify them using 'language labels' or 'symbols' (known as *discovery learning*) (Macnamara, 1982). After age 3, the learning process of children evolves through asking questions and obtaining clarifications of relationships between old concepts and new concepts (known as *reception*) (Coffey et al., 2003). The integration of relevant prior knowledge to learn new information is known as *meaningful learning* (Novak & Gowin, 1984). Meaningful learning is the opposite of *rote learning*. In rote learning, learners simply memorise facts with little or no effort to relate new knowledge with relevant existing information in learner's cognitive structures. The meaningful learning process is encouraged in every field to build expertise, due to its support for constructing effective knowledge structures. Figure 2.1 demonstrates the distinction between rote and meaningful learning.

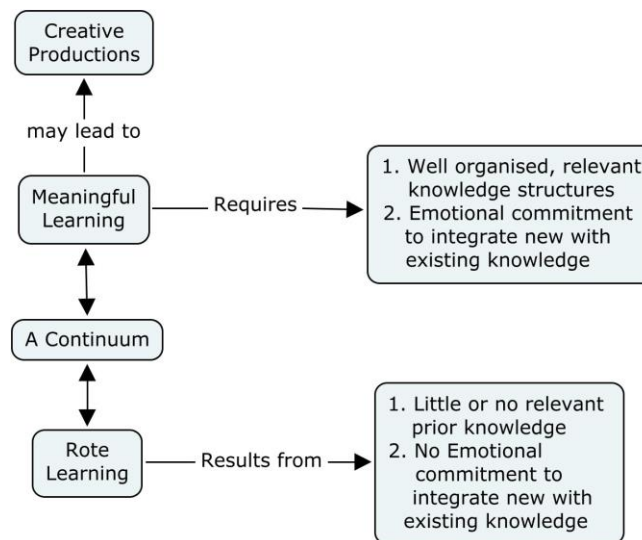


Figure 2.1: Distinction between rote and meaningful learning (Novak & Canas, 2006)

The hierarchical structure of concept maps supports assimilation theory by identifying the general concepts held by the learner prior to introducing more specific concepts (Coffey et al., 2003). In addition, identification of relations between concepts, and more progressively, identification of cross-links between concepts, involves a high level of cognitive performance such as in the 'evaluation' and 'synthesis' levels of Bloom's taxonomy (Bloom et al., 1956) since learners begin to see that every concept could be related to every other concept (Edmondson, 2005). Therefore, concept mapping as an aid to higher levels of meaningful learning is known to support Bloom's taxonomy of educational objectives (Edmondson, 2005; Novak, 1993; Novak & Canas, 2006).

The early view of long-term memory organisation states that “*information is stored in hierarchies, with more general information at the top and more specific information below*” (Ausubel & Robinson, 1969). Later, a model proposed by Anderson (1974) suggests that “*long-term memory organisation is a propositional network where information is presented as propositions and their interrelationships are store in a network-like fashion*”. By integrating these two models, the current view of long-term memory is established as a ‘*hierarchically organised network model of propositions*’ (Ormrod, 2011). Information is organised and represented by concept maps (i.e. hierarchical and propositional network structure) in the same fashion as it is in human memory. This supports the powerful connection of concept maps with the way that humans store and retrieve information in their long-term memory.

2.3 Applications of Concept Mapping

The ability of concept maps to facilitate meaningful learning promotes the application of concept maps as a learning and assessment tool. According to Nesbit & Adesope (2006), during 1997 to 2006, more than 500 empirical studies indicated that concept maps could be successfully utilised for learning and teaching.

The primary method to assess understanding is to assign scores to the components and structure of the concept maps constructed by students (known as the *traditional scoring system*) (Novak & Gowin, 1984). This system assigned 1 point for each valid proposition, 5 points for each level of the hierarchy, 1 point per branch, 10 points for cross-links and 1 point for each specific example (Novak & Gowin, 1984). Other approaches extended the traditional scoring system of Novak & Gowin (1984) by introducing different weights (Pearsall et al., 1997). Scoring system provides information about the creator’s knowledge structure in terms of concepts, relations and hierarchy. However, the *aggregate score* calculated using the traditional scoring system does not support the identification of ‘invalid’ ideas held by students (Ruiz-Primo & Shavelson, 1996).

Alternatively, Kinchin et al. (2000) proposed a qualitative approach to analyse students’ level of understanding through the structure of students’ constructed concept maps. Their work denoted the structure of concept maps as ‘spoke’, ‘chain’ or ‘net’ (Figure 2.2). The ‘spoke’ type denotes a radial structure where all the concepts are linked directly to the ‘root’ without connecting to each other (Figure 2.2 (a)). The ‘chain’ type is a linear sequence of understanding which connects only those that are immediately above and below (Figure 2.2 (b)). The ‘net’ type is the highly integrated structure of the concept map with a hierarchical network structure, demonstrating a deep understanding of the topic (Figure 2.2 (c)). Thus, the learners who possess ‘net’ type concept maps were identified as meaningful learners.

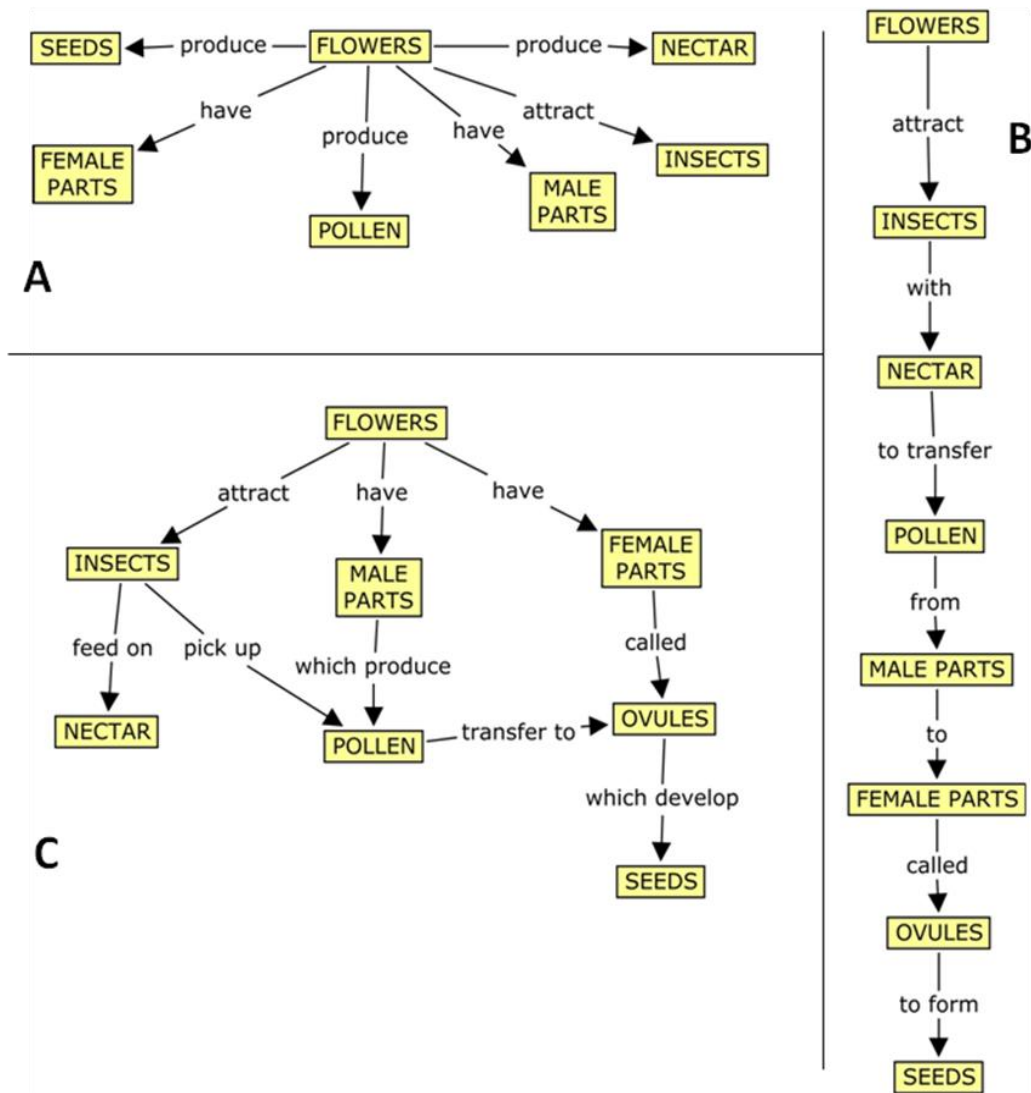


Figure 2.2: Concept map structures a) spoke b) chain c) net (Kinchin, et al., 2000)

In addition to the demonstration of conceptual understanding, concept mapping is an established technique to assess ‘conceptual change’, which measures the evolution of conceptual understanding over time. Results of a study which requested 68 students (science majors) who enrolled in an introductory Biology course to draw four concept maps of the same topic over four weeks duration showed a significant improvement in ‘structural complexity’ and ‘knowledge restructure’ by identifying larger number of concepts and cross links (Pearsall et al., 1997). Some longitudinal studies in science subjects revealed how conceptual changes occur in subsequent years and how concept maps could be a highly sensitive tool for measuring changes in knowledge structure (Novak, 1990).

Previous studies observed that learners often cannot transfer information learned in one context into another context. This notion is identified as ‘misconception’ where faulty or invalid knowledge structures are transferred into the other context. A study conducted at Harvard University asked graduates to explain ‘*why we have seasons?*’. Results demonstrated that 21 out

of 23 failed to give the correct answer since participants were unable to relate the knowledge into the real context even though they have theoretically learned about ‘*seasons*’ (Novak, 2002). The ability of concept maps to identify misconceptions was demonstrated using a study on Chemistry concepts – ‘acids’ and ‘bases’ with 34 students. Students’ concept maps were compared with corresponding expert maps followed by a MCQ test and a clinical interview. The results revealed that individual’s concepts are not consistently coincident with expert maps and students retained everyday concepts rather than scientific concepts (Ross & Munby, 1991).

Concept maps can also be used as a tool for *collaborative knowledge construction* (Boxtel, Linden, & Kanselaar, 2000; Coleman, 1998; Roth & Roychoudhury, 1993), particularly with the use of computer-based tools such as IHMC CMapTools (Canas, Hill, et al., 2004). Previous studies suggested that small groups working together to collaboratively construct concept maps had greater learning outcomes in many contexts (Canas et al., 2001). A study was conducted with 40 students to learn about ‘Electricity’ concepts. Student groups who received concept maps showed more discussion on Electricity concepts, collaboratively elaborated conflicts and reasoning than the control group who received an alternative resource (Boxtel et al., 2000).

Collaborative concept mapping in High School Physics with 29 students was evaluated both in groups and as individuals (Roth & Roychoudhury, 1993). The groups who had better concept maps demonstrated good discussions during collaborative activity and produced rich linkages and hierarchy. Similar effects occurred when the students in the group produced maps individually. However, the authors claimed that group work did not always increase individual performance even though students seem to be well motivated (Roth & Roychoudhury, 1993). Collaborative problem solving using concept mapping showed the importance of having group discussions to advance belief (Coleman, 1998).

Conversely, other studies found that collaboration in concept mapping did not appear to be beneficial for students. A study of O’Neil, Chung & Herl (1999) with 30 ninth grade students concluded that no significant correlation was found between ‘team processes’ and ‘team outcomes’. They described team processes as collaborative construction of a knowledge map by a team through communication between team members based on *adaptability, communication, coordination decision making, interpersonal* and *leadership*. Another study, which facilitated biological problem solving using concept mapping with 40 students, showed no statistically significant difference between students who mapped concepts cooperatively and those who mapped concepts individually (Okebukola, 1992). A meta-analysis on this area can be found in a work of Basque & Lavoie (2006) which reviewed 39 prominent published research papers on collaborative concept mapping based on number of participants, activities, discipline, whether computer-based or paper-based and collaborative or individual. However, this meta-analysis did

not conclude a consistent effect of collaborative concept mapping or individual concept mapping.

In general, students obtained significantly higher mean scores on achievement tests and positive effects on attitudes when utilising concept maps in the classroom than those who do not undertake concept map activities (Nesbit & Adesope, 2006; Pearsall et al., 1997). In addition to the learning improvement, concept maps as an educational tool explicitly changed the conceptual understanding held by students. They assisted to modify invalid knowledge structures (i.e. *misconceptions*) held by students (Novak, 2002).

Apart from the benefits to learners, concept maps are also a useful, reflective tool for teachers. Concept maps can be used to create ‘conceptually transparent’ curriculum planning where domain concepts are organised in advance without introducing redundancies or lessening its organisation (Ferry et al., 1997; P. A. Martin, 2008; Novak, 1998). Additionally, concept maps act as ‘*advance organisers*’ to organise the knowledge of the document to be learned as an overview (Novak & Canas, 2006). A study by Ferry et al. (1997) utilised concept maps to assist 69 preservice teachers to plan curriculum in Science. This study found that the concept mapping process helped preservice teachers create a more integrated view of subject matters and improved their curriculum planning skills (Ferry et al., 1997).

Concept maps demonstrated ‘knowledge elicitation’ in the form of the difference between novice and experts by comparing semantic and structural features of concept maps constructed by experts and novice learners. According to a study with 23 counsellors, experts possessed ‘schemas’ of ‘deep-level’ psychological principles to create concept maps with less concepts than the novices (J. Martin, Slemon, Hiebert, Hallberg, & Cummings, 1989). Therefore, concept maps demonstrated an ability in externalising experts’ tacit knowledge which is often difficult to articulate well to others through other media such as writing a book or delivering a lecture (Novak & Canas, 2006). In addition to the application areas discussed in this review, a comprehensive review of literature pertaining the use of concept maps can be found in the work by Coffey et al. (2003).

2.4 Knowledge Organisation versus Text Representations

This thesis focuses on knowledge organisation through concept maps as a supplementary technique to sequentially-structured lecture slides. This section starts the discussion with a background study of benefits on utilising general knowledge organisation techniques over text representations, and later in the discussion focuses on the specific interest in concept maps over PowerPoint lecture slides.

A study of 43 participants in the form of knowledge maps (i.e. treatment group, $n = 22$) or text representations (i.e. control group, $n = 21$) studied 1500 words on the 'autonomic nervous system' and rated their motivation, anxiety and concentration out of 10. Afterwards, students completed a performance test. The results demonstrated significantly higher scores for recall, subjective concentration and motivation in the treatment group compared to the control group (Hall & O'Donnell, 1996).

An extensive study compared 38 high and low ability students from Psychology classes who were taught using knowledge maps or traditional text (Patterson, Dansereau, & Wiegmann, 1993). The results showed that low ability students taught using knowledge maps performed better than the control group. However, high verbal ability students did not vary based on the medium of the study. Additionally, students recalled more central ideas when they learned using knowledge maps. Another study with 74 undergraduates on Biology found that the choice of utilising knowledge maps or traditional text also depends on learner's prior knowledge. Students with low prior knowledge learned most when the lecture was accompanied by knowledge maps. Higher prior knowledge students had no effect on the medium while for some students, utilising knowledge maps might create conflicts with their existing knowledge structures (Lambiotte & Dansereau, 1992). These findings were supported by early experiments where Bower et al. (1969) suggested that memory of hierarchically organised items was two to three times more effective than a random list. However, the research findings discussed here can vary based on the 'type' of text material (Angela M. O'Donnell, Dansereau, & Hall, 2002). For instance, (Patterson et al., 1993) found that students recalled more ideas from a passage on 'cocaine' than on 'alcohol'. Further, (A. M. O'Donnell & Dansereau, 2000) found different results with material on 'probability theory' and 'autonomic nervous system'.

An important aspect of designing teaching material is to facilitate students in understanding the important concepts of the domain and the relationships between new and existing knowledge. However, current instructional methods such as lecture slides which are developed on top of presentation frameworks are not supportive for knowledge organisation due to their point-based nature (Brandt et al., 2001; Kinchin et al., 2008; Kinchin et al., 2000). There has been a long-

standing criticism of the cognitive style of PowerPoint presentations where it is believed to weaken verbal and spatial reasoning when utilised as a lecture delivery mechanism (Tufte, 2003). Irrespective of its pedagogical value or cognitive ability, point-based lecture slides are embedded in the culture of teaching and hence, widely popular across the world (Apperson et al., 2008; Atkins-Sayre et al., 1998; Bartsch & Cobern, 2003; Frey & Birnbaum, 2002; Sugahara & Boland, 2006). Therefore, the knowledge organisation of lecture slides through concept mapping as a supplementary technique is discussed in the thesis. This argument is supported by previous studies of Bradley et al. (2006) and Kinchin et al. (2008). As stated in their work, it is challenging for teachers to explicitly express complex knowledge structures using point-based lecture slides. Therefore, it is unpredictable for teachers to perceive how students' will interpret and reconstruct the knowledge. It is possible for learners to construct false hierarchies that are not intended by teachers. Figures 2.3 and 2.4 depict an overview of such scenario and a real example in the field of Zoology respectively (Bradley et al., 2006; Kinchin et al., 2008).

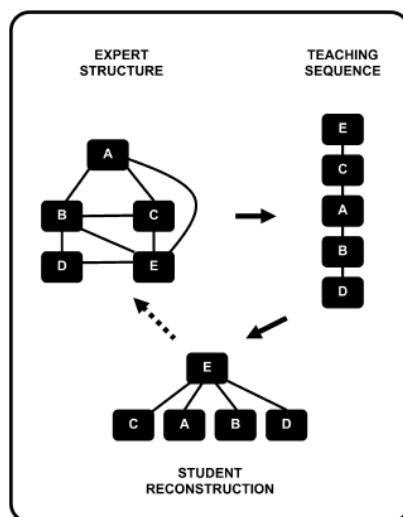


Figure 2. 3: Overview of transformation cycle (Kinchin, et al., 2008)

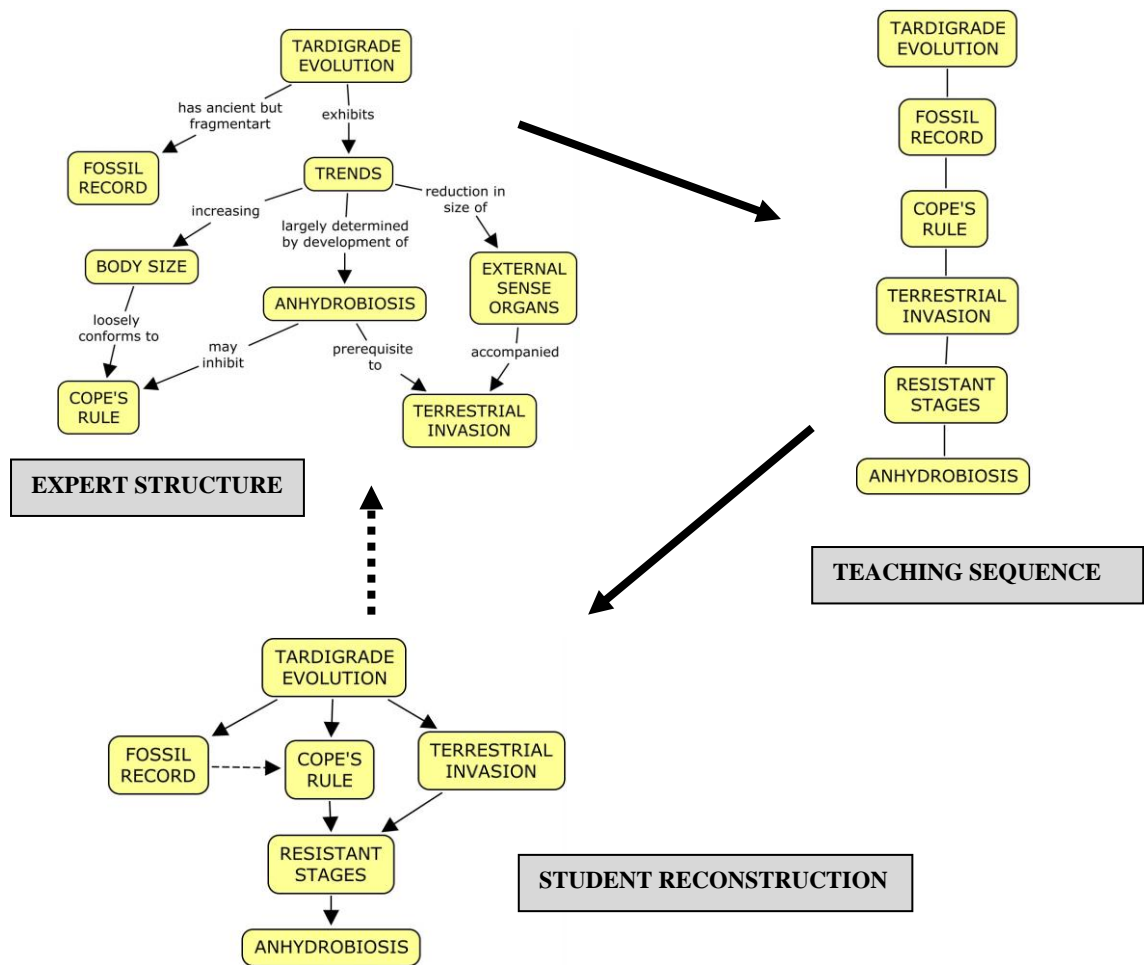


Figure 2.4: An example of a transformation cycle (Bradley, et al., 2006; Kinchin, et al., 2008)

According to Figure 2.4, the integrated knowledge structure of a subject expert ('expert structure') is transformed into teaching space using the sequence of slides ('teaching sequence'). Therefore, students often interpret the information presented early in the sequence as the most important. For instance, if the teacher introduces a new concept at the beginning (e.g. 'Cope's rule'), there is a possibility that a learner could interpret it as directly connected to the main idea without fully knowing *how* this concept connects to the main idea of the topic (depicted in 'student reconstruction'). Therefore, as stated in the work of Kinchin (2006a), the combination of lecture slides and concept maps contributes to an epistemologically balanced teaching approach.

Further, as a result of an in-depth comparison between concept mapping and lecture slides developed using PowerPoint, Kinchin et al. (2008) proposed Table 2.1 which depicts an overview the interactions between PowerPoint, the teachers and the students.

Table 2.1: Comparison between concept mapping and PowerPoint (Kinchin, et al., 2008)

PowerPoint	Concept mapping
Promotes single perspective	Promotes multiple perspective
Promotes linear knowledge structure	Promotes integrated knowledge structure
Reflects objectivist epistemology	Reflects constructivist epistemology
Focus on content	Focus on learning
Assessment-led	Understanding-led
Promotes passivity	Promotes dialogue
Promotes rote learning	Promotes meaningful learning

In contrast to the benefits for learners, developing a conceptual overview of a lecture provides a reflective tool for the teacher. Assessing the quality of educational materials can be challenging and, in Computer Science Education, where concepts can be represented in many forms not limited to Standard English, the ability to generate concept maps from materials provides one way of assessing whether the learning design has been transferred successfully to teaching materials.

According to this review, utilising knowledge organisations in contrast to text representations has noticeable benefits to both learners and teachers (Bower et al., 1969; Hall & O'Donnell, 1996; Kinchin et al., 2008; Lambiotte & Dansereau, 1992; Angela M. O'Donnell et al., 2002; Patterson et al., 1993). On some occasions, knowledge organisation techniques can be applied as an alternative to text representations, such as for learners with low verbal ability or low prior knowledge. In other situations, they can be provided as supplementary to traditional techniques, such as combining concept maps and lecture slides in the classroom for an epistemologically balanced teaching approach (Kinchin et al., 2008).

2.5 Concept Maps as Scaffolding

Expert maps are the concept maps prepared by domain experts (e.g. teachers) with the intention to serve for *scaffolding* purposes to facilitate learning.

Scaffolding

Scaffolding is the support given during the learning process with the intention of helping the student achieve his or her learning goals. This support is provided when the learner needs it, specifically when the learner lacks knowledge to complete a task or has misconceptions. The concept of ‘scaffolding’ was originally introduced in the context of adults assisting children in acquiring knowledge or solving problems in informal learning environments (Wood, Bruner, & Ross, 1976). Scaffolding is grounded in the Social Constructivism Theory of Vygotsky (1978) and his popular concept known as the *Zone of Proximal Development* (ZPD) (Beed, Hawkins, & Roller, 1991; Dabbagh, 2003). Due to the effectiveness of scaffolding in informal learning, the concept of scaffolding was adopted to formal education.

According to the works of Bean & Steven (2002), Dennen (2004) and Lepper et al. (1997), scaffolding supported learners both cognitively and affectively. Cognitively, scaffolding techniques offer strategies for problem solving and provide relevant information for learning. Affectively, scaffolding creates engaging environments for learners to achieve goals that they cannot achieve on their own, allowing learners to become more confident and increase positive feelings towards learning (Bean & Stevens, 2002; Lepper et al., 1997).

According to Hannafin et al. (1999), scaffolding is classified into four different categories based on its function: *conceptual scaffolding*, *metacognitive scaffolding*, *procedural scaffolding*, and *strategic scaffolding*. Conceptual scaffolding provides learners guidance about what to consider (Bean & Stevens, 2002). Metacognitive scaffolding supports learners to regulate their own learning (Davis, 1996). Procedural scaffolding helps learners to use resources and tools provided by the learning environment (X. Ge & Land, 2003). Strategic scaffolding provides tips and techniques for learners (Azevedo, Cromley, & Seibert, 2004). Primarily, scaffolding is provided by the teachers (called *scaffolders*) for learners to accomplish tasks that they cannot complete on their own (Lepper et al., 1997). An illustrative model of scaffolding is shown in Figure 2.5 (Hogan & Pressley, 1997).

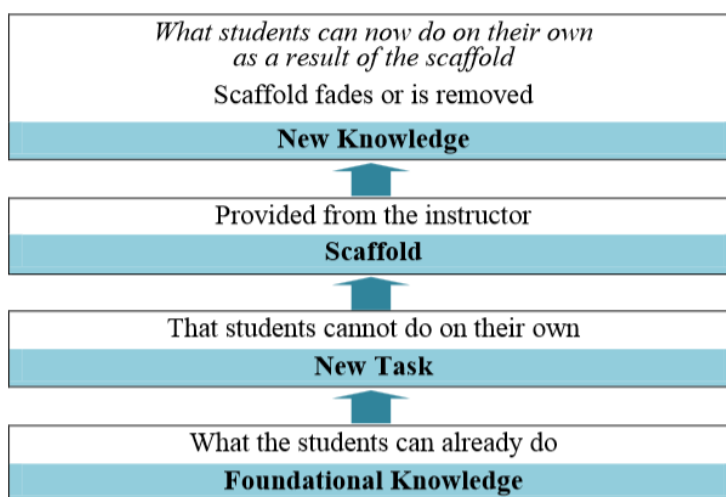


Figure 2.5: Illustrative model of scaffolding

Computer-based scaffolding is introduced into the learning environments as a supplement to human scaffolding (Beed et al., 1991). Computer-based scaffolding can take many forms such as multimedia resources, tasks and guidance (known as *instructional scaffolding*) (Jumaat & Tasir, 2014; Sawyer, 2006). Unlike human scaffolders, computer-based scaffolding is beneficial as it can be made available all the time, particularly in online learning environments. Recent research demonstrated the success of utilising scaffolding to improve distance learning (Bean & Stevens, 2002; Quintana et al., 2004). Table 2.2 presents examples of instructional scaffolding types proposed by Alibali (2006).

Table 2. 2: Types of instructional scaffolding (Alibali, 2006)

Scaffold	Ways to use scaffolding in an instructional setting
Advance organisers	Tools used to introduce new content and tasks to help students learn about the topic (e.g. Venn diagrams, flow charts, organisational charts, outlines, rubrics)
Cue cards	Prepared cards given to individuals or groups of students to assist in their discussion about a particular topic or content area (e.g. vocabulary words)
Concept maps and mind maps	Maps that show relationships. Partially completed maps for students to complete or completed maps. Students create their own maps based on their current knowledge of the task or topic
Examples	Samples, specimens, illustrations and problems
Handouts	Prepared handouts that contain task- and content-related information, but with less detail and room for student note taking
Hints	Suggestions and clues to move students along
Question stems	Incomplete sentences which students complete

The effect of computer-supported scaffolding has been studied widely, finding that instructional scaffolding demonstrates a positive effect on learning (Azevedo et al., 2004; Chang et al., 2001; Davis, 2000; Ruiz-Primo et al., 2001).

An early study of computer-based metacognitive scaffolding using a scaffolding software called *Reading Partner* with 25 college students to facilitate reading comprehension demonstrated statistically significant results on both reading comprehension and essay writing tasks using scaffolding compared to the students who did not receive scaffolding (Salomon, Globerson, & Guterman, 1989).

A scaffolding software tool (called *IdeaKeeper*) has been designed to facilitate students to engage in ‘online inquiry’ (Zhang & Quintana, 2012). This tool supports inquiry planning, information search, analysis and synthesis. A study based on *IdeaKeeper* assigned 16 sixth grade College students into two groups. The first group (treatment) performed online inquiry using the scaffolding tool and the second group (control) completed the online inquiry task in a regular setting using Web. The analysis of 80 screen videos of students’ activities and conversations demonstrated that *IdeaKeeper*-supported online inquiry was more integrated, efficient, continuous, metacognitive, and focused.

Azevedo et al. (2004) investigated the effect of different scaffolding conditions on a study of the ‘circulatory system’. A total of 51 undergraduates were randomly assigned to one of the three scaffolding conditions; adaptive scaffolding (AS), fixed scaffolding (FS) or no scaffolding (NS). Students in both AS and FS groups received an overall learning goal. Additionally, the AS group received dynamic and adaptive support from a tutor. The FS group received a list of domain-specific questions to help them understand the circulatory system. Students in the NS group did not receive any support. Results were significantly better in the AS group. However, there was no significant difference between the fixed scaffolding and no scaffolding groups.

The effectiveness of procedural scaffolding in the development of students’ group discourse levels and learning outcomes was measured using a study with sixty undergraduate and graduate students on ‘products sales’ (Huang, Wu, & Chen, 2012). Students were randomly assigned into either a procedural scaffolding group or a non-procedural scaffolding group. Participants in the procedural scaffolding group used smart phones with built-in cameras to learn new knowledge by scanning a barcode embedded in paper-based learning materials. The learning outcome between pre- and post-test scores demonstrated that experimental groups who received procedural scaffolding achieved better learning outcomes than the control group in terms of group discourse levels, group learning, and individual learning.

Expert Maps as Scaffolding

Novak & Canas (2006) introduced the idea of *expert skeleton maps* which provide a learner with a small concept map prepared by an expert in the domain. Typically, this map would have 6-10 concepts with proper linking words and a good hierarchy. The learner is required to extend this concept map by around three times the number of concepts contained in the expert skeleton map (Evens et al., 1997; Leelawong & Biswas, 2008; Angela M. O'Donnell et al., 2002; Olney et al., 2011).

Chang et al. (2001) and Ruiz-Primo et al. (2001) introduced the idea of 'fill-in-the-structure' in expert maps (Naveh-Benjamin et al., 1995). A prominent study in the area of scaffolding compared three techniques: '*construct-by-self*', '*construct-on-scaffold*' and '*construct by paper-and-pencil*'. This experimental study involved 48 students from a 7th grade Biology class, with participants randomly assigned into three groups. The '*construct-by-self*' group constructed their own concept maps and received evaluation result and feedback from the system. The '*construct-on-scaffold*' group received an incomplete map with list of concepts, relations, and hints as suggestions. Conversely, the '*construct by paper-and-pencil*' group drew their own concept maps on paper using only their own judgement (Chang et al., 2001). The analysis of their post-test scores revealed that '*construct-on-scaffold*' was significantly better than those of the '*construct-by-self*' and '*construct by paper-and-pencil*' groups. The scores of both the '*construct-by-self*' and '*construct by paper-and-pencil*' groups had the same effects on Biology learning. A qualitative analysis of students' opinions on different versions of the concept mapping technique revealed that students favoured having scaffolding and feedback during map construction.

Although '*construct-on-scaffold*' significantly outperformed the other two approaches based on performance scores, it has drawbacks as described in (Chang et al., 2001). It restricts the flexibility of map construction, particularly not supporting students with different learning styles or skills. Therefore, students in the '*construct-on-scaffold*' group mentioned that they will use concept mapping in short- or medium-term learning. According to Chang (2001), this may be due to the fact that 'fill-in-the-structure' becomes a kind of constraint that limits students' map construction, particularly when students' knowledge becomes more sophisticated. However, the '*construct-by-self*' group were willing to utilise concept mapping in long-term learning. An alternative to the short/medium-term learning is proposed using an approach called '*scaffold-fading*' (Chang, Sung, & Chen, 2002), where students are given '*construct-on-scaffold*' at the beginning and progressively weaned off pre-constructed maps for long-term learning gain (Chang et al., 2002).

A similar type of study conducted with 152 high school Chemistry students measured the effect of ‘*degree of directedness*’, ranging from ‘high-directed’ to ‘low-directed’ (Ruiz-Primo et al., 2001). The ‘high-directed’ concept mapping task provided more information including structure, linking phrases and concepts. Therefore, ‘high-directed’ tasks were similar to ‘*construct-on-scaffold*’ in the study of Chang et al. (2001) which provided nodes and/or link labels to ‘fill-in-map’. The ‘low-directed’ tasks were similar to that of construct from scratch with no scaffolding provided. The concept mapping task model with the ‘degree-of-directedness’ and the ‘amount of information’ is illustrated in Figure 2.6.

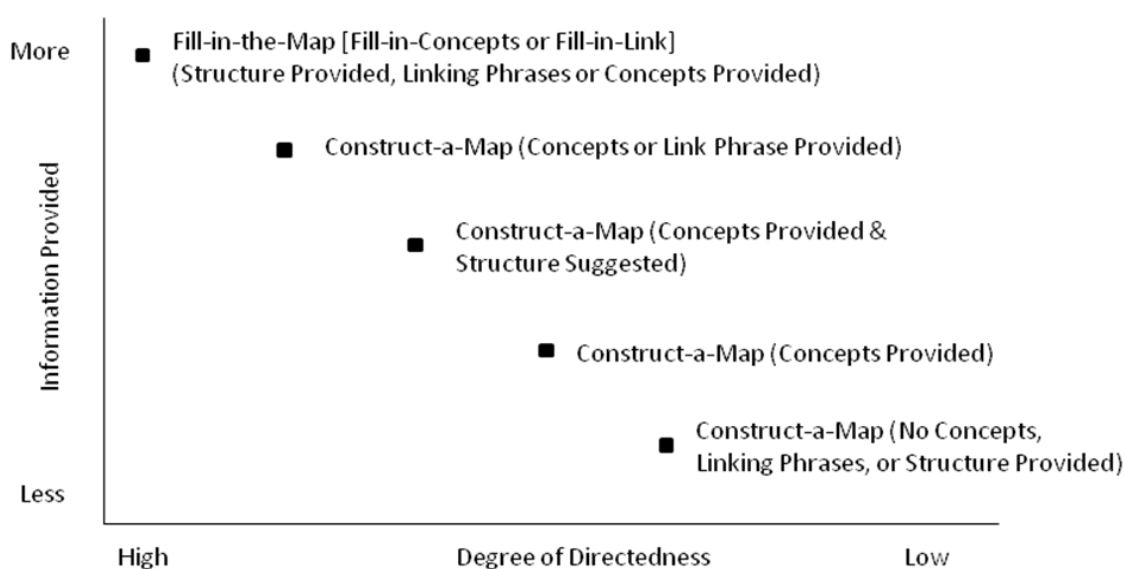


Figure 2.6: Concept mapping tasks model (Ruiz-Primo, 2004)

The results of this study demonstrated that ‘*high-directed*’ student group scores reached close to the maximum possible (i.e. 12 points). This was similar to the study by Chang et al. (2001).

In addition to providing elements of concept maps as scaffolding to construct or fill-in concept maps, expert constructed maps, or auto-generated concept maps which can act as expert maps, are provided to students for studying or problem solving (Bagno & Eylon, 1997; Hauser et al., 2006; Horton et al., 1993; Angela M. O'Donnell et al., 2002; Valerio & Leake, 2012).

A study assigned 102 participants into four treatment conditions to construct concept maps; from scratch (*map-generation*), using list of concepts (*concepts-provided*), from spatially arranged concepts (*concepts-arranged*), or study a concept map (*worked-out map*), after reading a text on ‘ethical and biological issues in human embryo research’. The control group had no mapping tasks. Results demonstrated that *worked-out map* and *map-generation* groups exhibited learning improvement. This study demonstrated that both studying from expert maps and constructing maps from scratch were equally significant for learning (Hauser et al., 2006).

A meta-analysis of 19 studies on science concept learning analysed a sample size of 95 students. In 15 out of 19 studies, students manually constructed concept maps. In another 3 studies, the teacher prepared concept maps (i.e. *expert maps*) for students to use in learning. In the other study, both teacher and students prepared the concept maps. The results reported that there was little difference in the effectiveness of teacher-prepared versus student-prepared concept maps in improving students' achievement. In teacher-prepared maps, the average students' achievement improved from the 50th to the 71st percentile while in the student-prepared maps, the average students' achievement improved only from the 50th to the 66th percentile (Horton et al., 1993).

Concept Maps as Scaffolding for Problem Solving

The particular interest of this thesis is on the utilisation of concept maps as scaffolding for problem solving, and specifically in answering questions. With the goal of utilising concept maps as scaffolding for answering questions in an online environment, relevant literature is reviewed in this section.

In the educational context, questioning has been widely used to engage and motivate students, focus attention, guide learning, and provide the opportunity for practice and self-assessment (Dillon, 1988; Hunkins, 1972; Wilen, 1986). There are different types of questions that can be constructed within the education context. Among them, most researches construct questions based on educational taxonomies such as Bloom's taxonomy (Bloom et al., 1956; Hunkins, 1972). The cognitive domain of Bloom's taxonomy contains six levels of educational objectives: knowledge, comprehension, application, analysis, synthesis, and evaluation (Table 2.3).

Table 2.3: Bloom Taxonomy of educational objectives (Bloom, et al., 1956)

Level	Description	Keywords
Knowledge	Recall information	Define, describe, identify, list, select, recognise, state, outline
Comprehension	Understanding or interpretation	Explain, distinguish, give an example, interpret, paraphrase, summarise, rewrite, translate
Application	Apply knowledge to new situation	Apply, construct, demonstrate, discover, manipulate, modify, prepare, produce, relate, show, solve, use
Analysis	Distinguishes between facts	Analyse, compare, contrast, diagram, differentiate, discriminate, distinguish, identify, illustrate, outline, relate, select, separate
Synthesis	Put parts together to create new meaning	Categorise, combine, create, design, explain, generate, modify, organise, plan, rearrange, reconstruct, relate, reorganise, revise, rewrite, summarise
Evaluation	Make judgements	Conclude, criticise, evaluate, justify, support

In addition to Bloom's taxonomy, questions can be classified as closed-ended (e.g. *yes-no*, *MCQ*) and open-ended questions, factual and opinion questions, and lower and higher order thinking questions (Dillon, 1988). Graesser et al. (1992) suggested 18 different question types using an extensive analysis of questions are being asked by both students and tutors (Table 2.4).

Table 2.4: Question classification by (Graesser, et al., 1992)

Question category	Abstract specification
1. Verification	Is X true or false? Did an event occur?
2. Disjunctive	Is X, Y, or Z the case?
3. Concept completion	Who? What? When? Where?
4. Feature specification	What qualitative properties does X have?
5. Quantification	How much? How many?
6. Definition questions	What does X mean?
7. Example questions	What is an example of a category?
8. Comparison	How is X similar to or different from Y?
9. Interpretation	What can be inferred from given data?
10. Causal antecedent	What state causally led to another state?
11. Causal consequence	What are the consequences of state?
12. Goal orientation	What are the goals behind an agent action?
13. Procedural	What process allows an agent to reach a goal?
14. Enablement	What resource allows an agent to reach a goal?
15. Expectation	Why did some expected event not occur?
16. Judgemental	What value does the answerer give to an idea?
17. Assertion	A declarative statement that indicates the speaker does not understand an idea.
18 Request/Directive	The questioner wants the listener to perform some action.

A study was conducted with 40 pre-degree Biology students. Students who had experience in concept mapping for six months (called '*concept mappers*') were compared with a control group who did not have prior concept mapping experience. The results showed that *concept mappers* were significantly more successful in solving biological problems than the control group (Okebukola, 1992).

In another study with 250 students in the area of 'Electromagnetism', students solved problems related to *Maxwell's equations* while constructing concept maps gradually according to an integrative model (*solve, reflect, conceptualise, apply and link*). Results showed a significant improvement in recall, conceptual knowledge and problem solving when compared to a control group using a drill and practice approach (Bagno & Eylon, 1997).

2.6 Summary

This chapter provided the background of the research from the theoretical perspective of concept mapping in terms of meaningful learning and human memory. The chapter included a discussion on empirical studies which adopt concept maps in class room education to assess several aspects such as student understanding, misconceptions, conceptual change, collaborative learning and problem solving. In addition, this chapter reviewed the benefit of utilising knowledge organisation techniques (e.g. concept maps) over sequentially-structured text representations (e.g. text books, lecture slides). The chapter discussed the idea of ‘expert maps’ and applications of utilising them as a scaffolding technique.

Even though the purpose of introducing expert maps as scaffolding is to overcome the issues associated with manual construction of concept maps by learners, the widespread adoption of expert maps as a technique is hindered by the additional workload involved on domain experts, requiring detailed articulation of the domain model, and creation of complex concept maps.

Therefore, this thesis proposes an approach to produce expert maps for learning with minimal human interventions. As a consequence, an approach for automatically generating concept maps from various text sources is reviewed in the next chapter (Chapter 3).

Chapter 3

Concept Map Mining: A Review

This chapter presents the background works of the concept map mining (CMM) process. The chapter includes an overview of CMM process, the CMM systems and methods developed and their evaluation methodologies. Finally, application areas of CMM within the educational context are presented.

3.1 Overview of Concept Map Mining

According to the definition by Villalon & Calvo (2008), the concept map mining (CMM) process can be expressed as the proper extraction of a concept map from a document (D). This process has three steps;

1. Concept extraction (CE) and identifying the set of concepts (C)
2. Relation extraction (RE) and identifying the set of relations (R)
3. Topology extraction (TE) and identifying a generalisation (G) of the set of concepts

CE must be the first step of the process, because C defines R and G. Every term used to describe a concept or a relationship must appear in the document, therefore D itself defines all potential words (or phrases). This idea can be formalised by defining a document as a triplet $D = C_d, R_d, G_d$ where C_d corresponds to all the concepts, R_d corresponds to all the propositions, and G_d corresponds to the levels of generalisation expressed in the document.

The next stage is to identify the subsets C, R and G that are a good summary of D. Figure 3.1 illustrates the concept map mining process.

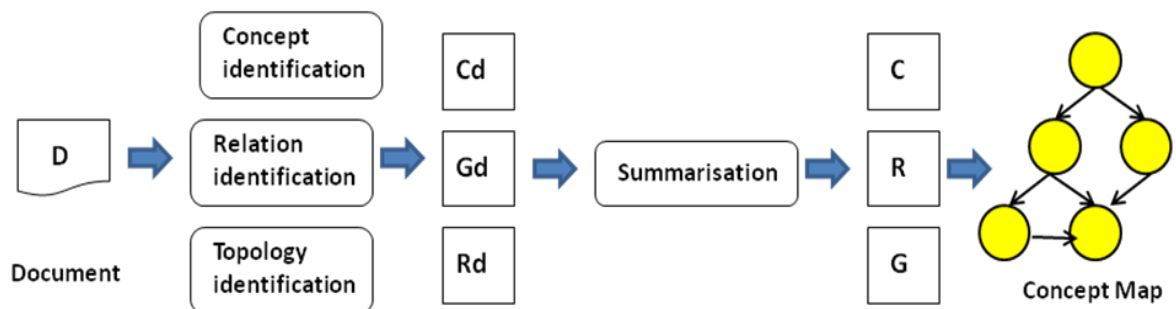


Figure 3.1: Concept map mining process (Villalon & Calvo, 2008)

3.2 Concept Map Mining Systems and Methods

This section starts with the background studies of concept, relation, and hierarchy extraction, and ranking based on the CMM definition by Villalon & Calvo (2008). Subsequently, *concept map* extraction as a whole from various text sources in terms of underlying methods and their level of automation is presented.

Concept Extraction

The majority of automated concept extraction approaches rely on statistical-based information retrieval (IR) methods (Manning, Raghavan, & Schütze, 2008; Salton & Buckley, 1988; Salton & McGill, 1986). These methods utilise distributional features such as *term frequency* (Salton & McGill, 1986), *C-value/ NC-value* (Frantzi, Ananiadou, & Mima, 2000), and *co-occurrence* analysis (Clariana & Koul, 2004). C-value/NC-value method incorporates statistical and linguistic information of text (Frantzi et al., 2000). Co-occurrence analysis considered that two terms have a strong relations if they co-occur frequently in a document or a domain (Manning et al., 2008). Term frequency calculates the number of occurrences of a term and if the term occurs more frequently, then it is considered as important in the given domain. Generally, unnecessary terms are eliminated using *stop word* removal by applying Porter's stemming algorithm (Porter, 2006) or *lemmatisation* techniques (i.e. converting words into their base form, e.g. *processes* => *process*, *interesting* => *interest*) (Klein & Manning, 2003a, 2003b) and *tokenisation* (Y.-H. Tseng et al., 2010) before calculating term frequency (Cimiano, 2006). Stop words are words which are filtered out before or after processing of natural language text (e.g. *a*, *the*, *is*).

Conversely, natural language processing algorithms obtain part-of-speech tags or parser trees of text to identify nouns, noun phrases, adjectives and then to identify possible concepts by manually constructing 'text patterns' (Table 4.11, page 82) (Cimiano, 2006). In addition, language models (e.g. *n-gram* model) are utilised to identify consecutive sequence of *n* items to form domain concepts.

Relation Extraction

The majority of relation extraction approaches utilise machine learning methods. Machine learning methods can be of several forms such as supervised learning, semi-supervised learning, and unsupervised learning. Feature-based supervised learning approaches for relation extraction employ Maximum Entropy models by combining various features such as lexical, syntactic parse trees, and dependency trees (Kambhatla, 2004). This approach was evaluated with Automatic Content Extraction (ACE) test data using 1679 instances of relations. ACE data includes *entities* (e.g. persons, organisations, and locations), *mentions* (e.g. names, nominal

expressions or pronouns) and relations between entities. Results indicated a very high *precision* and low *recall* compared to the baseline approach. Precision is the fraction of retrieved components (usually by the computer-based algorithms) that are relevant and the recall is the fraction of the relevant components that are successfully retrieved by the algorithms. The success of feature-based methods depends on the ability to define good feature set which can be obtained through extensive experimentations.

In contrast, kernel-based supervised learning methods do not require a pre-defined set of features. Bunescu & Mooney (2005) utilised the shortest path between two ‘named entities’ in the same sentence to obtain the relationship by extracting the dependency graph of the sentence. A named entity is a pre-defined category of elements in text such as the names of *persons*, *organisations*, *locations*, *expressions of times*, *quantities*, *monetary values* and *percentages* (Sang & Meulder, 2003). An evaluation using 97 documents of ACE newspaper corpus based on ‘CFG dependency parser’ produced better results and outperformed related works which used tree kernels to define the dependency graph (Culotta & Sorensen, 2004; Zelenko, Aone, & Richardella, 2002). However, kernel-based methods are more computationally complex than feature-based methods.

Even though supervised learning approaches are more efficient, they are restricted to pre-determined relations or patterns (e.g. *company-founded_by*, *author_of-book*) defined when labelling the *training set*. A training set is a labelled document (usually by humans) which considers as the input to the supervised learning algorithm to learn the classification. For instance, a training set labels the relations such as *located-in (University of Adelaide, Adelaide)* and *located-in (Monash University, Melbourne)* manually and applies the learned algorithm to a *test set* (e.g. new document) to classify unknown data (e.g. *located-in (CMU, Pittsburgh)*) (Bach & Badaskar, 2007). Supervised learning methods are less applicable to open relations and hence, difficult to extend to extract new relations. Open relation extraction is the task of identifying relationships between two or more entities in text without pre-determined patterns like *company-founded_by*, *author_of-book*. Even though some works do not use hand-labelled patterns, they rely on external databases such as *WordNet* (Miller, Beckwith, Fellbaum, Gross, & Miller, 1990) as discussed by Snow et al. (2005).

Semi-supervised learning approaches generally use examples such as *author_of-book*, *company-founded_by* to label data (known as ‘seed examples’) and induce patterns from text. These patterns are then applied to unlabelled data to extract new relations (known as *bootstrapping*). The *DIPRE* (Dual Iterative Pattern Relation Extraction) system applied this technique to learn *author_of-book* pairs from the Web using a web crawler (Brin, 1999). A repository of 24 million web pages was used to evaluate the system, learning roughly 346 patterns. These

techniques are constrained by the large amount of data required for training purposes. *Snowball*, a quite similar approach to DIPRE, learns *organisation-location* relations from a collection of more than 300,000 newspapers. However, *Snowball* limits its usage since it relies on ‘named entities’ such as *organisation* and *location* (Agichtein & Gravano, 2000).

An alternative to predefined relation types and seed examples was introduced in *TextRunner*, an unsupervised approach to open information extraction (OIE) (Banko, Cafarella, Soderland, Broadhead, & Etzioni, 2007). This approach automatically discovered relations without human intervention of hand-labelled data and learned examples through a ‘self-supervised learner’. These learned examples were then applied to a test corpus of 9 million web documents, and extracted 7.8 million well-formed tuples. A tuple is an ordered list of elements. In the context of relation extraction, ‘tuple’ is considered as ‘entity-relation-entity’. From a randomly selected subset of 400 tuples, 80% were considered to be accurate according to human evaluators. A comparison with the state-of-the-art system called *KnowItAll* (Etzioni et al., 2005) showed a reduced error rate by 33%. When compared to other methods discussed above, unsupervised learning approaches require minimal human intervention. However, unsupervised learning approaches like *TextRunner* depend on NLP tools such as a dependency parser to automatically label examples. This process requires good quality natural language sentences to reuse relation extraction systems like RelEx (Fundel, Kuffer, Zimmer, & Miyano, 2007).

The extraction of relations in the form of concept-relation-concept triples was presented in some recent research (Dali & Fortuna, 2008; Rusu, Dali, Fortuna, Marko, & Dunja, 2007). Triple extraction proposed by Rusu et al. (2007) assumed no prior knowledge about text (e.g. named entities). Instead, their approach utilised grammatical structures of English sentences to define heuristic rules. This approach compared three popular syntactic parsers; Stanford/OpenNLP (Klein & Manning, 2003a, 2003b), Link grammar parser (Sleator & Temperley, 1991) and Minipar (Lin, 2003) to identify subject-verb-object (SVO) triple in English sentences. This thesis reuses some of the heuristics defined by Rusu et al. (2007) to extract SVO from sentences which possess the pattern (*Root (S (NP_subtree) (VP_subtree)*).

The work proposed by Dali & Fortuna (2008) extracted all possible ordered combinations of three tokens (i.e. *subject*, *verb* and *object*) in English sentences to train a ‘support vector machine’ (SVM) using human annotated triples. The training of the SVM model utilised over 300 features including length of sentence, number of stop words, subject-verb distance, part-of-speech of candidate terms, and the depth of the tree. In binary classification system (either ‘positive’ or ‘negative’), 39% of precision and 47% of recall has been achieved for a test set of 100 sentences from Reuters’ news corpus. This work was limited since it considered all combinations of three tokens. Therefore the computation was substantially increased with the

length of sentences, so if sentence has 10 tokens, possible triple combinations are $10 * 9 * 8 = 720$.

Hierarchy Extraction

In the context of concept mapping, hierarchical organisation implies the arrangement of concepts according to their importance, imposing the *generalisation-specialisation (is-a)* relation. An automated approach to learn hypernym (*is-a*) relations was discussed in a work of Snow et al. (2005). This approach utilised an external database called *WordNet* (Miller et al., 1990) to learn such relations, in contrast to the hand-written regular expression patterns typically used in traditional approaches (Cimiano, 2006). The learned hypernym pairs using *WordNet* can then be used to discover new relations from text. Learning hyponym relations without relying on external databases was proposed by Hearst (1992). In Hearst's work, six most commonly used *lexico-syntactic patterns* were introduced (e.g. *such as*, *and other*, *including*). The example in Figure 3.2 shows the identification of hierarchical relationships with the use of lexico-syntactic patterns (i.e. *and other*).

Pattern: *NP [, NP]*[,] and other NP* (Note: *NP* denotes 'Noun Phrase')

Example: *Java, C++ and other programming languages are designed to communicate instructions to a computer.*

According to the lexico-syntactic patterns, *programming languages* identifies as the 'parent'. *Java* and *C++* identifies as 'children', creating the 'generalisation/specialisation (is-a)' relationship.

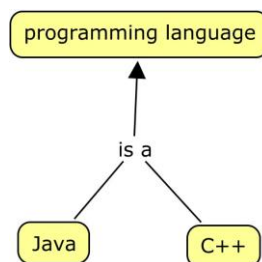


Figure 3.2: An example for generalisation/specialisation (is-a) relation

While lexico-syntactic patterns are utilised to learn 'parent-child' relations, *co-occurrence* analysis is widely used to identify 'siblings'. As discussed in the work of Riloff & Shepherd (1997) and Sanderson & Croft (1999), patterns such as '*and*' (e.g. *Presentation, application, and session layers ...*) reflect that the participating terms co-occur in the local context, and hence, identify as 'sibling' relations. Later research extended these patterns to identify other relation types such as '*part-of*' using following patterns (Cimiano, 2006).

Pattern: X's Y => System's function

Pattern: X of a/an Y => Operating system of a computer system

These types of patterns were used as 'seed examples' in unsupervised learning methods to learn new hierarchical relations using the *bootstrapping* techniques (Manning et al., 2008). Even though applying pattern-based approaches such as lexico-syntactic patterns and co-occurrence demonstrate high accuracy, they limit their usage due to their lack of availability in any corpus, resulting in low recall (Cimiano, 2006). More specifically, lecture slides lack these type of patterns due to their point-based nature and the use of sentence fragments in preference to complete sentences.

Ranking

According to the definition by Villalon & Calvo (2008), the final stage of CMM is ranking (or summarisation) of the extracted knowledge according to the importance of concepts. According to the work of Afantenos et al. (2005), ranking can be categorised into either *statistical-based* or *graph-based*. As already discussed within 'concept extraction', statistical methods utilise techniques such as term frequency, inverted document frequency (Manning et al., 2008; Maynard, Li, & Peters, 2008; Salton & McGill, 1986), C-value/NC-value method (Frantzi et al., 2000) and Latent Semantic Analysis (Deerwester, Dumais, Furnas, Landauer, & Harshman, 1990; Villalon & Calvo, 2009) to rank the extracted knowledge.

In addition, graph-based methods utilise features specific to graph theory. In graph theory, directed graphs usually define two separate measures of degree centrality, namely *in-degree* and *out-degree*. Accordingly, *in-degree* is a count of the links directed to the node (i.e. concept) and *out-degree* is the number of links that the node directs to others (Diestel, 2005) (see Figure 4.8).

A study in Indiana University and Institute for Human and Machine Cognition (IHMC), Leake et al. (2004) suggested that "*candidate topological and layout factors might influence decisions of which concepts are most topic relevant*". This study considered four candidate models. The Baseline model considered map topology and layout as unimportant. The *Connectivity Root-Distance Model* (CRD) model suggested in-degree, out-degree (i.e. degree of centrality), and the *proximity* (i.e. distance from the *root* concept) influence the importance of particular concept. Unlike the CRD model which considered immediate neighbours into account, the *Hub Authority and Root-Distance Model* (HARD) model considered the full map and determined whether a node (i.e. concept) is a *hub*, *authority* or *upper node*. Node is defined as *hub* or *authority* when a particular node has multiple out-degree or in-degree respectively. The *Path Counter Model* (PC) model suggested the concepts that participate in more propositions are

more important than other nodes. Twenty participants were recruited to evaluate this study, where the participants were requested to answer 56 questions about 12 small concept maps. In each question, participants were presented with a concept map and two concepts selected from map and asked to examine ‘*which of the two concepts best describes the map’s topic?*’ or ‘*whether both described equally well*’. Results demonstrated that the layout of the map had no significant effect, however, the CRD model outperformed the HARD model when compared with human judgements. Therefore, in-degree, out-degree and proximity were concluded as influential structural factors to rank concepts in a concept map. To prevent domain knowledge from influencing participants’ decisions, this study replaced concept labels with artificial terms. Therefore, this approach lacked the domain knowledge to determine which concepts are most topic relevant (Leake et al., 2004).

Concept Map Extraction

This section discusses concept map extraction studies based on their underlying methods. The section starts with statistical methods such as *co-occurrence*, *term frequency*, *inverse document frequency* and the limitations of these methods are discussed. Afterwards, linguistic methods such as *syntactic parsing*, *part-of-speech tagging* and *language models* are explained and the discussion is concluded with the limitations of linguistic methods. In addition, concept map extraction using machine learning approaches such as *naive Bayes classifier*, *association rules* and *clustering* are described. The section is concluded with the benefits of having hybrid approaches to extract concept maps.

Research by Clariana & Koul (2004) defined a list of important terms in the Biological domain. The *co-occurrence* of this pre-defined list was compared with the ‘terms’ in students’ written summaries with less than 30 sentences and constructed an ‘aggregated proximity array’ by assigning ‘1’ when co-occurred and ‘0’ otherwise. Finally, a concept map-like *Pathfinder Network (PFNet)* representation (Schvaneveldt, 1990) was generated from the proximity array. *PFNet* lacked relation labels since this approach focused only on whether there is any association exists between terms and not ‘how’ they are connected. The ability of *PFNet* to capture important propositions from written text was moderate (Pearson $r = 0.69$) according to the human evaluators (Clariana & Koul, 2004).

Concept map extraction from ‘e-learning’ conference papers (between year 2001 and 2004) and journal articles (between year 1999 to 2004) as the data source provided a useful visual overview of the domain for researchers, particularly beginners in the field (N.-S. Chen et al., 2008). Their approach considered ‘keywords’ defined by the authors on research paper as ‘concepts’ and the distance between any of these two ‘keywords’ within the article as

‘relationship strength’. This approach hypothesised that “*the shorter the distance between two key words, the higher the relation between them*”. A 10-point Likert scale questionnaire was used to collect opinions from 2 e-learning domain experts. The experts agreed that relationships and overall fitness was up to 80% and concept ranking was 70% according to their professional knowledge. Similar to the work of Clariana & Koul (2004), this approach lacked ‘relationship labels’ and ‘hierarchy’ of concept maps. Alternatively, it provided ‘relationship strength’ between terms using a numeric value.

Concept maps mined from Chinese ‘news articles’ reflected the scientific knowledge contained in daily news stories (Y.-H. Tseng et al., 2010). This approach utilised a rule-based algorithm to extract key terms. Their algorithm was applied to a collection of 100 Taiwan news articles and obtained 10% improvement in the ‘accuracy’ compared to the baseline approach. The first k terms ($k = 30$) were ranked based on the *term frequency-inverse document frequency* measure and the association between the top k terms were calculated using co-occurrence analysis of nearly 25,000 Chinese news articles. Five assessors were recruited to judge the relevancy between selected 30 term pairs. Based on their judgement, 69% of the associated terms were relevant within the selected set of terms. This approach also lacked ‘relation labels’ and hierarchy of concept maps.

In general, statistical methods such as *term frequency*, *inverse document frequency* and *co-occurrence analysis* do not rely on domain knowledge or external resources. Therefore, statistical methods can be applied to any domain for knowledge extraction. However, statistical methods commonly suffer from probable semantic loss (Zubrinic, Kalpic, & Milicevic, 2012). These methods fail to identify and overcome semantic ambiguities like pronouns and determiners which are used as a substitute for a noun or noun phrase. Additionally, statistical methods are not able to detect synonyms (i.e. word with the similar meaning of another word) and homonyms (i.e. words that share the same spelling and pronunciation but have different meanings). Instead, statistical methods focus on the distribution of information within the corpus.

Alternatively, linguistic knowledge about text is utilised to extract concepts and relations. The widely used techniques of linguistic methods include syntactic parsing, part-of-speech tagging, dependency parsing, named entity tagging and language models (Cimiano, 2006). The important aspect of linguistic methods is that they do not depend on quantitative measures such as *term frequency*, but rely on the ‘grammatical structure’ of the text. In general, linguistic methods extract nouns (or compound nouns) as ‘concepts’ and verbs as ‘relations’. However, there can be nouns existing which are not ‘concepts’ in that particular domain. Additionally, there can be verbs which act as nouns in some domains (e.g. *gerund* verbs). In order to

overcome these issues, external databases such as *WordNet* (Miller et al., 1990) were utilised in linguistic-based methods (Alves et al., 2001). Though these types of external resources are available in domains like Biology and Medicine (Fisher, Wandersee, & Moody, 2002), external taxonomies are very limited in the domain under study (Computer Science).

Machine learning methods utilised classification systems such as a naive Bayes classifier (Zouaq & Nkambou, 2008), association rules such as fuzzy rules (S.-M. Chen & Bai, 2010; Lau et al., 2008) or clustering techniques to extract concept maps from text sources. Concept maps extracted from learners' historical testing records allowed visualisation of students' learning performance related to particular concepts of a given course. A study by Tseng et al. (2007) with 104 junior high school Physics students proposed the Two-Phase Concept Map Construction (TP-CMC) approach which enabled mining of association rules from testing records and their corresponding quizzes. This process constructed concept maps using the concepts extracted from quizzes and built the prerequisite relationships between these concepts using the students' testing records. Results showed that this approach is practical for analysing and visualising learners' testing records. The majority of other concept map extraction researches utilised a combination of machine learning, statistical and linguistic methods (known as 'hybrid' methods).

The recent advancement of the field of knowledge acquisition utilises hybrid methods where, in general, linguistic or machine learning methods for knowledge extraction and statistical methods for ranking the extracted knowledge (Lau et al., 2008; Olney et al., 2011; Valerio & Leake, 2006; Villalon & Calvo, 2009; Zouaq & Nkambou, 2008, 2009). In addition, some other works utilised linguistics methods for knowledge extraction while machine learning methods to build the concept maps (Alves et al., 2001; Wang et al., 2008).

TextStorm extracted relations between concepts as binary predicates (e.g. *John eats meat => eat(John, meat)*) using Natural Language Processing techniques and the *Cloud* system interactively completed the missing knowledge in concept maps by asking primitive questions from users (e.g. *Question: Define lions with the predicate 'is-a' => Answer: is-a (lion, animal)*) using machine learning methods (Alves et al., 2001). With a sample of 21 small text files of articles, manuals, educative texts, *TextStorm* achieved a correctness mean of 52% in extracting binary predicates. Although the approach operated independently from domain knowledge, *TextStorm* limits its usage since it depended on *WordNet* (Miller et al., 1990) for lexical verifications of the words. Further, this approach is limited to affirmative and declarative sentences.

The *Fuzzy Association Concept Mapping* (FACM) technique has been proposed to automatically extract concept maps from abstracted short texts (Wang et al., 2008). This technique utilised linguistic methods to extract propositions (in the form of ‘concept-relation-concept’ triples) from text and interactively refined the extracted proposition with the use of human recommendations using fuzzy set theory. This framework was evaluated with the Science Citation Index (SCI) abstract database and CNET news. The proposed FACM approach was compared with a baseline algorithm called ANNIE (A Nearly-New Information Retrieval System) based on the GATE platform (Cunningham, Maynard, Bontcheva, & Tablan, 2002) and obtained higher precision (87% for ‘abstracts’ and 83% for ‘news articles’) and higher recall (78% for ‘abstracts’ and 74% for ‘news articles’).

Fuzzy domain ontology-based concept map construction from ‘discussion forum topics’ employed linguistic-based *lexico-syntactic* patterns and statistical methods (Lau et al., 2008). A small scale study of 10 undergraduates posted discussions on the forums on a familiar topic. The system generated a concept map from the posts and invited 10 students to assess whether the extracted concept maps reflected their understanding. Participants used five parameters: *accuracy*, *cohesiveness*, *isolation*, *hierarchy* and *readability* using five scales ranging from ‘very poor’ to ‘very good’. The overall results demonstrated that *accuracy* and the *quality* of auto-generated concept maps were promising ($M = 4.3$, $SD = 0.61$).

Valerio and Leake (2006) assumed that the source document should be ‘well-written’ to apply existing linguistic techniques such as syntactic parsing and part-of-speech tagging. Their approach utilised *term frequency* to rank the extracted concepts according to importance. Although the proposed system extracted concepts and relations as nouns and verbs respectively, it lacked hierarchy extraction. With a sample of 80 documents selected randomly from STORM-LK knowledge model, this approach achieved 57% *precision* and 71% *recall* (Valerio & Leake, 2006).

TEXTCOMON (Text-Concept Map-Ontology), which is based on the *Knowledge Puzzle Ontology Learning Tool*, focused on extracting concept maps from text using machine learning and linguistic methods (i.e. parsing and lexico-syntactic patterns) (Zouaq & Nkambou, 2008, 2009). This approach converted extracted concept maps from text to domain ontology. These domain ontologies were then utilised in the knowledge modelling of intelligent systems. A corpus of 36 text documents containing approximately 30,000 words in e-learning domain was used for an evaluation. The system outperformed similar tools built previously for the same purpose and evaluated using the same corpus (e.g. TEXT-TO-ONTO (Cimiano & Volker, 2005)).

Concept map mining from a Biology text book (Olney et al., 2011) utilised the *SemNet* formulations (Fisher et al., 2002) to generate concept maps. In this approach, key terms were used as ‘start nodes’ of a triple, where the ‘end node’ could be either key terms or a complete proposition. The concept maps generated, which was consistent with *SemNet*, allowed comparison with thousands of Biological triples available online. They reused key terms existing in the ‘glossary’ and ‘index’ of text books and the test-prep study guides. After applying triple extraction algorithms to extract nearly 4400 triples, a manual categorisation was carried out to cluster them according to relationship types. Finally, statistical-based filters were applied to discard triples that were not suitable for concept map exercise generation. With the use of two experts having background in Biology and pedagogy, a Wilcoxon signed ranked test, pairing human and computer generated maps found that computer generated maps were more ‘accurate’ to be utilised as *expert maps* ($Z = 2.13, p < .03$). One of the benefits of having a ‘text book’ as the mining source is that text books contain grammatically complete sentences with minimal ambiguities (e.g. pronouns). This approach had limitations in that the auto-generated concept maps contained fewer links (approximately 3.5 times) than expert maps. Therefore, it was difficult to reuse them to compare with student constructed or modified maps. Additionally, this system failed to extract every triple from every sentence which resulted in a low *recall*.

Concept map mining from students’ written essays on the topic of ‘English as a global language’ helped visualise the concepts and relations included in essay form (Villalon & Calvo, 2009). This approach utilised grammar trees to identify concepts and Latent Semantic Analysis (LSA) for ranking. The system suggested a list of concepts from student essays (approximately 12 concepts per essay) and human annotators built concept maps from them, enabling the visualisation of the essay as a concept map. Even though the human-machine agreement (i.e. accuracy) was not greater than *inter-rater agreement*, the system reported promising results when compared to the related works in the literature. Inter-rater agreement is the agreement between human evaluators when more than one evaluator is involved in an evaluation task (Manning et al., 2008).

The hybrid method is intended to overcome the specific issues discussed under statistical and linguistic methods. Statistical methods suffer from probable semantic loss while linguistic methods are suited for *well-written* natural language text (Zubrinic et al., 2012). Additionally, linguistic methods extensively rely on external databases such as *WordNet* (Miller et al., 1990).

Table 3.1 compares the CMM systems discussed above based on underlying data source, methods and the features in the form of relation and hierarchy extraction. Each of the CMM systems were capable of extracting concepts.

Table 3.1: Comparison of concept map mining systems

Systems	Data source	Method	Relation	Hierarchy
TextStorm (Alves et al., 2001)	Text file	Hybrid (linguistic and machine learning)	Yes	No
(N.-S. Chen et al., 2008)	Academic articles	Statistical (<i>term frequency and relationship strength</i>)	Yes, but no relationship labels	No
(S.-M. Chen & Bai, 2010)	Answers to questions (both correct and incorrect)	Machine learning (<i>association rules</i>)	Yes, but no relationship labels	No
ALA-Reader (Clariana & Koul, 2004)	Written summaries of students	Statistical (co-occurrence)	Yes, but no relationship labels	No
(Lau et al., 2008)	Discussion forums	Hybrid approach (<i>lexico-syntactic patterns</i> and statistical method)	Yes (taxonomy relations only)	Yes (fuzzy taxonomy of relations)
(Lee et al., 2009)	Incorrect answers to exam questions	Machine learning (association rules)	Yes, but no relation labels	No
(Olney, 2010; Olney et al., 2011)	Text book	Hybrid (Natural language processing and statistical methods)	Yes	No
TP-CMC (S.-S. Tseng et al., 2007)	Learner's historical testing records	Machine learning (Fuzzy association rules)	Yes, but no relation labels. Instead used weighted relations	No
(Y.-H. Tseng et al., 2010)	Chinese news articles	Statistical method (rule-based extraction algorithm & co-occurrence analysis)	Yes, but no relation labels	No
(Valerio & Leake, 2006)	Well-written text document	Hybrid method (linguistic and term frequency)	Yes	No
(Valerio & Leake, 2012)	Well-written Document	Hybrid method (linguistic and text mining)	Yes (disjoint partial maps also available)	Yes
(Villalon & Calvo, 2009, 2011)	Student essay	Hybrid (linguistic and Latent semantic analysis)	Yes in the later work (Villalon & Calvo, 2011)	Creates manually

TEXTCOMON (Zouaq & Nkambou, 2008)	Text documents	Hybrid (machine learning and linguistic method)	Yes	No
(Wang et al., 2008)	Abstracted short text	Hybrid (Linguistic and fuzzy set theory)	Yes (Interactively built with human users)	No

In addition to the prominent works related to the concept map mining from various text sources within the educational context, there are quite a few preliminary studies which extracted information from *lecture slides* to form concept maps, dependency graphs, semantic networks, or mind maps. Among them, Gantayat & Iyer (2011), Kasinathan et al. (2013) and Person et al. (2012) utilised automated approaches to extract dependency graphs, mind maps and concept maps respectively.

Dependency graph extraction from lecture slides utilised hybrid method including statistical and machine learning techniques. However, the approach of Gantayat & Iyer (2011) is limited in extracting concepts only. Similarly, the mind map extraction by Kasinathan et al. (2013) was focused on extracting concepts only. The concept map extraction by Person et al. (2012) utilised linguistic techniques such as dependency parsing to extract concepts and relations. Table 3.2 summarises these researches based on their construction manner, inputs, outputs, methods and functionalities.

Table 3.2: Knowledge organisation from lecture slides

Approach	Construction manner	Knowledge source	Knowledge organisation technique	Method	Components	Performance
(Gantayat & Iyer, 2011)	Automatic	Computer Networks & Operating Systems lecture slides in NPTEL courseware repositories	Dependency graph with pre-requisites and follow-up topics	Hybrid (<i>TF-IDF</i> & <i>A priori</i> algorithm)	- Concepts - No relations labels & hierarchy	Querying the courseware repository demonstrated 40% of answers were correct
(Ono, Harada, & Shimakawa, 2011)	Manual	Lecture slides of Programming	Semantic network	Manual	- Concepts & Hierarchy - No relation labels	28% agreement with humans
(Kasinathan et al., 2013)	Automatic	Lecture slides	Mind map	Structure-based algorithm (Cannot guarantee that structure is same in all	- Concept - No relation & hierarchy	Not clear

				sources e.g. font size of title ≥ 34)		
Guru (Person et al., 2012)	Automatic	Bulleted facts in Biology	Concept map	Linguistic (Syntactic dependency parsing)	- Concept & relation - No hierarchy	Concept map generation algorithm not specifically included evaluation results

Based on the background work reviewed, there are many tools or techniques available to extract concept maps or similar knowledge representations from semi-structured or unstructured text. However, to the authors' knowledge, there are no studies to this point, which have developed a fine-grained text analysis to extract concept maps from *lecture slides*.

3.3 Evaluation of Concept Map Mining

Although various tools and techniques have been developed to extract concept maps from text, a standard evaluation framework to measure the quality of generated concept maps is lacking (Villalon & Calvo, 2008). Therefore, comparison between similar studies is challenging since different research studies were evaluated differently. Additionally, the evaluation of either automated or manual knowledge acquisition is highly subjective, based on the individuals involved in the process. Due to the lack of a consistent evaluation framework, various researchers proposed different evaluation methodologies. According to Zouaq & Nkambou (2009), these different methodologies can be categorised into three dimensions: *structural*, *semantic* and *comparative*.

Structural

In the concept mapping perspective, the typical approach of 'structural evaluation' involves assigning scores to the components or structure of the concept maps under study (known as the *traditional scoring system*) (Kinchin et al., 2000; Novak & Gowin, 1984; Pearsall et al., 1997; Ruiz-Primo & Shavelson, 1996) (discussed in Section 2.3). This approach is proven to be a time-consuming process in assessing maps in large scale (Coffey et al., 2003).

Structural features to evaluate 'domain ontologies' was proposed by Zouaq & Nkambou (2009) using four metrics: *Class Match Measure* (CMM), *Density Measure* (DM), *Betweenness Measure* (BEM) and *Semantic Similarity Measure* (SSM). The total score from all these measures were calculated to rank the ontology based on given search terms. Accordingly, Class Match Measure could be used to match search terms either fully or partially with 'classes' in ontology. The Density Measure included details of particular concept such as number of subclasses, siblings, relationship with other concepts and attributes (attributes are not usually considered in concept

mapping). The Betweenness Measure indicated the extent to which a concept lies on the path between others, enabling the degree of centrality. Finally, the Semantic Similarity Measure calculated the semantic proximity between search term and corresponding matches in the ontology.

Semantic

Semantic evaluation considered expert maps as a *gold standard* to compare other concept maps constructed manually or generated automatically (Ruiz-Primo & Shavelson, 1996; Villalon & Calvo, 2008). Manual comparison of concept maps is effort-demanding, and hence, automated approaches were proposed by Park & Calvo (2008) and Herl et al. (1997). Park & Calvo (2008) converted the triples of concept maps into Semantic Web form as RDF or RDFS to obtain the exact match or similarity score between concept maps under comparison. This technique will be reused in this thesis. Instead of the RDF form, the approach used in this thesis utilises XML-based concept map elements.

Conversely, when expert maps do not exist for comparison, domain experts judge the validity of concept map elements using binary classification as either *relevant* or *non-relevant* based on their knowledge and experience (Manning et al., 2008). This judgement is then compared with information retrieved using computer algorithms, assessing '*how well computer and human agree on a component or structure as relevant*'. This is usually measured using *precision* and *recall*.

However, semantic evaluation involves subjectivity of judgement. In order to overcome this, more than one human evaluator is used and the agreement between human evaluators is calculated using *inter-rater agreement*. In general, most studies involved *two* (Olney et al., 2012; Olney et al., 2011), *three* (N.-S. Chen et al., 2008; Leake et al., 2004; Villalon & Calvo, 2011) or more evaluators (Clariana & Koul, 2004; Y.-H. Tseng et al., 2010; Villalon, Calvo, & Montenegro, 2010).

The *Kappa statistic* (or Cohen's Kappa) is the most popular measurement to calculate *inter-rater agreement*. As a rule of thumb, 0.8 is taken as 'good' agreement and values between 0.67 and 0.8 are taken as 'fair' agreement. When the agreement is below 0.67, the corpus under study believed to be uncertain for human annotation (Manning et al., 2008). If the human experts have 'fair' or 'good' agreement on judgements, this value is compared with the accuracy obtained from human to machine comparison (known as *human-to-machine agreement*). Generally, as proposed by Hearst (2000), '*machine extractions are acceptable if human-to-machine agreement is equal or higher than the inter-rater agreement*' (Villalon & Calvo, 2009).

An alternative approach to judge the validity of concept map elements was proposed by Olney et al. (2011) using three dimensions: *coverage*, *accuracy* and *pedagogy*. This study involved *two* human evaluators to rate 60 automatically extracted concept maps compared to *gold standard* maps extracted from the work by Fisher et al. (2002). The rating of the dimensions was scaled into four items. For instance, rating scale for ‘coverage’ assigned scores as shown below.

- 1 - The map covers the concept
- 2 - The map mostly covers the concept
- 3 - The map only slightly covers the concept
- 4 - The map is unrelated to the concept

Based on these ratings, human-to-machine agreement using mean scores was obtained as 2.47, 1.87 and 2.53 for *coverage*, *accuracy* and *pedagogy* respectively.

A similar approach to evaluate concept maps using five dimensions (*accuracy*, *cohesiveness*, *isolation*, *hierarchy* and *readability*) was proposed by Lau et al. (2008). This approach used a five point Likert scale for the ratings (e.g. from 1 - *very poor* to 5 - *very good*). An overall mean score of 4.3 and standard deviation of 0.61 was achieved from 26 domain concepts and 97 relationships arranged in 3 hierarchy levels.

Comparative

Comparative evaluation compares automatically extracted concept maps using various tools. The comparison is considered as valid when concept maps are evaluated using the same corpus. *TEXT-TO-ONTO*, an ontology learning tool from text (Cimiano & Volker, 2005) was compared with *TEXTCOMON* (Text-Concept map-Ontology), which extracts concept maps from text to transform into domain ontologies (Zouaq & Nkambou, 2008, 2009). The comparison utilised four domain ontologies generated from each tool using the same corpus. Their comparison dimensions were *concept existence*, *richness* and *interconnection levels*. However, ‘comparative’ evaluation is not widely used within the concept map mining context due to the lack of tools developed to extract concept maps from the same form of knowledge sources.

Apart from the three dimensions (*structural*, *semantic* and *comparative*) discussed above, automatically generated concept maps were evaluated based on the application context or their intended tasks (Brank, Grobelnik, & Mladenic, 2005). Producing a good result from the intended task does not necessarily suggest that the generated concept maps are also good unless

they are separately evaluated for the quality and validity. Therefore, a separate evaluation is required before they can be applied to a problem domain. The next section describes various applications of CMM, particularly within the educational context.

3.4 Concept Map Mining Applications in the Educational Context

Concept map mining applications within the educational context can be divided into two categories. In the first category, concept maps extracted from educational materials such as text books, lecture slides and academic articles can be utilised to create concept map activities to improve the learning experience. The second category extracts concept maps from learners' data (e.g. historical testing records, answers to questions) or tasks (e.g. written essays, forum discussions) to identify students' understandings and misconceptions.

Since 1987, concept maps (or conceptual graphs) have been utilised to model the domain knowledge of expert systems (Regoczei & Hirst, 1988, 1992) with the intervention of human experts. With the introduction of CMM techniques, auto-generated concept maps are utilised to model the domain knowledge of intelligent tutoring systems (ITS) such as *Guru* and *Betty's Brain* (Leelawong & Biswas, 2008; Person et al., 2012). *Guru* extracted concept-relation-concept triples from bulleted facts in lecture materials of high school Biology to construct concept map activities in an intelligent tutoring environment (Person et al., 2012). Students were expected to fill-in-the-blanks of skeleton maps extracted from CMM techniques as shown in Figure 3.3.

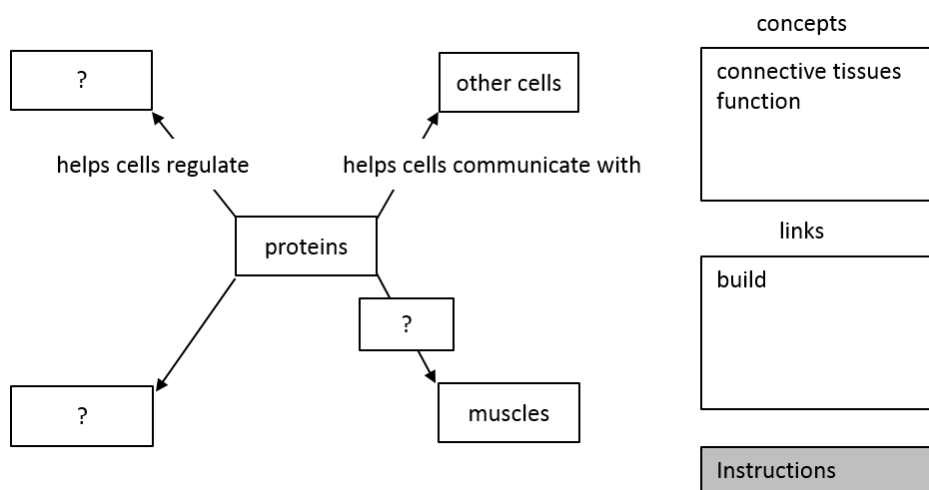


Figure 3.3: Concept map activity of *Guru* (Person, et al., 2012)

Person et al. (2012) conducted a study with 32 Biology students learning the required content using either *Guru* or human tutoring. The correlation between the time spent on concept map activities and the learning gain was measured. In the first cycle of concept map activities (out of

two cycles), time spent on the map was positively correlated with the learning gain. However, there was no significant correlation between concept map activities and learning gain in cycle 2. One of the key issues of this approach was that concept map generation algorithm relied heavily on domain experts to assess the extracted triples, which lessens the degree of automation.

Betty's brain is another prominent ITS which uses concept maps to 'learn by teaching' to a computer agent (Leelawong & Biswas, 2008). A study was conducted with 45 fifth-grade students to learn about the topic, *River ecosystem*. Students taught a computer agent called *Betty* using a concept map and asked questions to see how Betty had learned the concepts taught by the student. Finally, Betty was tested with quizzes provided through the system (a mentor agent called Mr. Davis). If Betty was not successfully answering system quizzes, the mentor agent provided hints to debug and correct the concept map ('*Ask Mr Davis*'). The performance scores of post-tests of three groups were compared. The control group involved learning by being taught (pedagogical agent). The first treatment (i.e. baseline) group performed learning by teaching to Betty and the second treatment group performed learning by teaching with feedback on self-regulated learning (SRL) strategies and domain content. Results indicated that the two learning by teaching groups (with concept maps and other tools) performed significantly better than the control group.

Concept maps generated from text books have been utilised for domain modelling of ITS and to generate student exercises (Olney, 2010; Olney et al., 2012; Olney et al., 2011). The question generation from concept maps utilised either individual triples or multiple triples to impose a degree of difficulty in generated questions. Individual triples produced simple question types like *hints* (e.g. *what do X do?*) and *prompts* (e.g. *Is X true or false?*). Multiple triples generated three question types (*contextual verification*, *forced choice* and *casual chain*). For instance, the contextual verification type was derived from 'generalisation/specialisation (*is-a*)' relations by providing the context and asking a question (e.g. *Cats have a tail. Is that true for dogs?*). Questions generated from concept maps of Biology text books were evaluated by *three* human experts based on five dimensions: *question type*, *relevance*, *fluency*, *ambiguity* and *pedagogy*. For instance, human experts rated 'question type' as either 'target question type' or 'type of generated question and the target question is different'. Each expert rated approximately 90 questions from five question types discussed above. Inter-rater agreement between judges was calculated, giving a majority of the agreement scores as satisfactory (Cronbach's $\alpha \sim 0.8$). However, reliability score for 'question type' was poor at $\alpha = 0.43$.

In contrast to the creation of learning activities using concept maps mined from educational materials, concept map mining algorithms were applied to students' data or tasks to extract concept maps to identify students' understanding and misconceptions. CMM from student

essays for cognitive visualisation considered as a reflective activity of individual's writing (Villalon & Calvo, 2009, 2011; Villalon et al., 2010). A total of 43 essays on the topic of 'English as a global language' written by students in University of Sydney were used for the experiments. Villalon & Calvo (2009) stated that the first stage of the experiments demonstrated an inter-rater agreement of 57% among *two* annotators which is quite low in comparison with other works. This occurred since annotators used their own interpretation, rather than those in the original essay when defining relationships. In a later study, the annotating procedure was modified and achieved higher reliability scores such as 77% for lexical precision and 85% for taxonomical precision (Villalon & Calvo, 2011). The cognitive visualisation through concept mapping from student essays provided feedback to improve writing prior to submitting the assessments.

Concept maps drawn by 111 college students were mined to identify learning preferences of students (Yoo & Cho, 2012). A total of 112 concepts were used in study in the categories such as class room learning (e.g. listening, discussion), learning tools (e.g. notes, text books), facility (e.g. web, library, writing centre). Students were requested to draw a concept map answering the question '*how do you learn in a college class?*'. The approach used data mining techniques such as *frequent concept mining* to identify which concepts are frequently using in students' maps and *sub-concept map mining* to observe which sub-concept structures are commonly used. The results demonstrated that students prefer the traditional way of learning by stressing the concepts '*me, teacher, notes, lectures, textbooks*' and associations such as *teacher-notes, teacher-me, me-notes*.

An application to develop *Intelligent Concept Diagnostic System* (ICDS) for teachers to identify student misconceptions and knowledge gaps through the analysis of students' data was proposed by Lee et al. (2009). This work automatically constructed concept maps from learners' wrong answers to the exam questions using *A priori* algorithm and association rules. Further, the system suggested remedial learning for students through the Remedial-Instruction Path (RIP) algorithm. Information loss was likely to occur when considering misconceptions and knowledge gaps (i.e. incorrect answers) regardless of what student already knows (i.e. correct answers), particularly when this information is utilised in adaptive learning systems. Therefore, Chen & Bai (2010) suggested extracting concept maps from both correctly and incorrectly answered exam questions.

The particular interest of this thesis is to utilise *auto-generated concept maps* as scaffolding to facilitate problem solving, specifically in answering questions in an online environments. Concept maps as scaffolding have already demonstrated positive effects on problem solving in the studies of Bagno & Eylon (1997), Okebukola (1992) and Bascones et al. (1985). A more

recent work of Valerio & Leake (2012) studied the effect of auto-generated concept maps for problem solving. Their study measured the reading comprehension skills in terms of both ‘speed’ (i.e. time taken to answer question) and ‘accuracy’ in answering questions. A group of 16 undergraduates and graduates were provided with 60 questions to answer using a text document, a concept map constructed manually or auto-generated maps. Results showed that providing auto-generated concept maps improved user speed in answering questions, for *well-written* documents whose size enabled generating a single concept map with a limit of 30 most important concepts. However, there was no significant difference between each resource on ‘accuracy’.

This thesis investigates an approach to provide the most relevant information to answer questions using concept maps generated from lecture slides. The motivation of this investigation is supported using an early study by Eylon & Reif (1984). Eylon & Reif (1984) suggested that “*higher levels of the hierarchy should preferentially contain information most important for the domain of tasks*”. This hypothesis was tested by comparing the effectiveness of two hierarchical knowledge organisations that contained the same knowledge, with one of them adapted to a set of given tasks. The study was conducted using 20 students enrolled in a high school Physics class. The students were randomly assigned into four treatment groups. First two treatment groups were given task-adapted information while one group received ‘stronger’ treatment and the other group received ‘weaker’ treatment. In strong treatments, the acquisition tasks were designed to reinforce the acquisition of the specified organisation. Similarly, the other two treatment groups (stronger and weaker) received the same information organised hierarchically, but not adapted to the tasks. After two weeks, students were given three types of tasks including writing a summary of previously studied information (*free-recall*), asking specific questions about learned facts (*cued-recall*) and problem solving tasks to make inferences on the basis of the learned facts. The results demonstrated that the former treatments (i.e. task-adapted information) performed significantly better than the latter treatments (i.e. hierarchical information without task adaptation) in the performance tasks. However, there was no significant difference between stronger or weaker treatments in each category. Even though the study by Eylon & Reif (1984) provides a basis for this research, it is not a feasible approach to organise the knowledge according to tasks *manually*. Therefore, this thesis takes a new perspective to develop a machine-based framework for task adaptation.

3.5 Summary

This chapter provided details of empirical studies which automatically extract concept maps from various text sources. The evaluation methodologies to determine the quality and validity of auto-generated concept maps were presented using three categories: structural, semantic and comparative evaluation.

This chapter divided CMM applications within the educational context into two categories. The first category of applications extracted concept maps from educational materials such as text books and lecture slides and utilised the extracted concept maps to create activities and assessments within the classroom. In contrast, the second category extracted concept maps from learners' data (e.g. historical testing records, answers to questions) or tasks (e.g. written essays, forum discussions) for visualisation to identify students' understandings and misconceptions.

Finally, a review on utilising concept maps as scaffolding for problem solving was included. This discussion concluded by drawing attention to the gap in the field. The next two chapters (Chapter 4 and 5) provide techniques developed to lessen the gaps which were identified using the review of literature.

Chapter 4

Concept Map Mining from Lecture Slides

This chapter presents the framework for automated concept map extraction from *lecture slides*. The extracted concept maps will be utilised as a positive alternative to expert maps to design pedagogical activities, particularly in the problem solving context.

To the authors' knowledge, there are no studies until this thesis which have implemented a full scale, fine-grained text analysis to extract concept maps from *lecture slides*. Related works in the field of research focused on extracting concept maps from various other text sources such as *text books* (Olney, 2010; Olney et al., 2011), *academic articles* (N.-S. Chen et al., 2008), *well-written text documents* (Alves et al., 2001; Valerio & Leake, 2006, 2012), and *student essays* (Villalon & Calvo, 2009). The underlying text sources of all these related works shared one key feature; they were all *well-written* with unambiguous and grammatically complete text.

The choice of text source in this research introduced a large amount of noise. Noisy text is defined as low quality text, which presents in natural language with typographic errors, colloquialisms, and grammatical issues which lower the accuracy of automated text processing using computers. Noisy text includes the use of sentence fragments, associated with bullet points, instead of complete sentences and increased ambiguity and confusing use of idioms. This makes lecture slides difficult for automated knowledge extraction, which led to incorporate *contextual features* to resolve syntactically, semantically missing and ambiguous elements in lecture slides.

This research generated all important elements of concept maps (i.e. *concepts, relations, relation labels, hierarchy, and cross-links*), in contrast to other CMM works which extracted components partially including topic maps (headings only) (Gantayat & Iyer, 2011), relations without labels (N.-S. Chen et al., 2008; S.-M. Chen & Bai, 2010; Clariana & Koul, 2004; Lee et al., 2009; S.-S. Tseng et al., 2007; Y.-H. Tseng et al., 2010), or concepts and relations without hierarchy (Alves et al., 2001; Olney, 2010; Olney et al., 2011; Valerio & Leake, 2006; Wang et al., 2008; Zouaq & Nkambou, 2008).

In the field of automated knowledge acquisition, 'concept extraction' from text is not a difficult problem to solve. It has already achieved reasonably good *precision* and *recall* using various methods such as statistical and linguistic (Atapattu, Falkner, & Falkner, 2012; Frantzi et al., 2000; Manning et al., 2008; Maynard et al., 2008; Salton & Buckley, 1988; Salton & McGill,

1986). However, *relation* extraction is still challenging, particularly when extracting ‘open relations’ (i.e. undefined relations) in specific domains. Therefore, the majority of machine-extracted concept maps in the educational context lacked relationships between concepts (N.-S. Chen et al., 2008; S.-M. Chen & Bai, 2010; Clariana & Koul, 2004), which made them ineffective in facilitating meaningful learning. The relation extraction mechanism of this research using natural language processing (NLP) techniques contributes to the advancement of the field of research.

The most common approach to assessing the quality of auto-generated concept map is measuring ‘*how well the computer and human expert agree on an element or structure as valid or relevant*’ (Villalon & Calvo, 2008). This is usually measured using precision and *recall*. However, in the educational context, particularly when considering lecture slides, the majority of knowledge presented is deemed relevant to the learner by design. Therefore, this research contributes the design of an evaluation methodology which considers the correlation between human and computer as a rank, with some highly important knowledge, averagely important knowledge, and others with low importance (Atapattu, Falkner, & Falkner, 2014b).

To measure the effectiveness of concept maps generated from lecture slides, the hypothesis was constructed as ‘*It is possible to develop auto-generated concept maps of lecture slides to strongly correlate with human constructed maps*’.

4.1 Design of the Concept Map Mining Framework

Prior to explaining the concept map mining framework (named CMMF), it is important to discuss the design considerations of the framework. The primary focus of the design of CMMF was to generate concept maps for teachers to obtain the conceptual overview of lecture slides, and further, utilised them to construct various concept map activities and assessments for learners. Therefore, the focus of the CMMF design will be explained from the teachers’ perspective.

Firstly, a software framework for educational context should be designed according to underlying learning theories (Tchounikine, 2011). Concept mapping as a learning technique is grounded in cognitive learning theories of Ausubel (Ausubel, 1968) as already discussed in Chapter 2.

Secondly, the design of a software framework to produce concept maps from existing resources should not introduce additional burden to the teachers. For instance, Kinchin (2006b) encouraged teachers to produce concept map-based lecture handouts to facilitate knowledge organisation. Additionally, Villalon & Calvo (2009) utilised human annotators to build concept

maps from a list of computer suggested concepts extracted from students' essays. It is beneficial for teachers to utilise their existing lecture slides with common formats, such as consistent headers and text for automated concept map generation. The structuring information that was used within CMMF generalises across any common lecture slide formats such as Apple KeyNote, OpenOffice and Latex. The CMMF was demonstrated in this thesis using Microsoft PowerPoint lecture slides as the basis material.

The only additional work involved in CMMF was defining the *topic* of the lecture using a meaningful, brief noun phrase excluding category headings like *Chapter 1, introduction, and review*. The topic of the lecture will be used as the 'root' or 'central concept' of the map. The selection of a single root node with a meaningful label represents the 'big picture' of the topic (Kinchin et al., 2008; P. A. Martin, 2008).

Some earlier works of CMM utilised external resources or external users to produce concept maps. For instance, Olney (2010) and Olney et al. (2011) utilised 'Biology triples available online' to produce relations. *TextStorm* (Alves et al., 2001) and Villalon & Calvo (2009) made use of external users to interactively build concept maps. The usage of external resources or users might improve the performance of the tool; however, such approaches are not accessible or utilised by a broader community. The design of CMMF did not utilise any external resources.

The *domain independent* approach of CMMF can be utilised across any discipline with availability of sufficiently mineable lecture slides. In this thesis, CMMF was demonstrated using the Computer Science domain within the University context. In order to be successful at University study, students must learn to independently acquire discipline knowledge and learn how to extract knowledge using varied sources of information, including lecture slides and text books amongst other sources, which can be well supported through concept map activities.

In order to develop new techniques to extract concept maps from lecture slides, this thesis analysed approximately 27 Computer Science courses, including those available within the local teaching context and from text book publishers. Each course contained approximately 22 lecture topics, which resulted in an analysis of a collection of nearly 600 slide sets. Additionally, each slide set contained approximately 40 slides. Therefore, the development of CMMF has analysed nearly 24,000 slides from CS courses.

The analysis of the corpus identified several issues associated with the Computer Science domain and the lecture slides under study, which provided challenges to applying the techniques developed in this thesis. Table 4.1 summarises these issues which will be referred throughout the thesis.

Table 4.1: Issues associated with the corpus

Category	Issue	Description
Text-related	Gerund verbs	English language often contains verbs in its <i>-ing</i> form that can function as a noun (e.g. <i>Software testing, mounting, Operating systems</i>)
Slide-related	Relations between bullet points	Generally, lacks logical relationships between preceding or succeeding bullet points
	Text	Semantically ambiguous text (e.g. pronouns, determiners), grammatically incomplete sentence fragments (e.g. noun phrases, verb phrases), colloquialisms
	Special text patterns	Lexico-syntactic patterns rarely exists in lecture slides
CS-related	Named entity	Generally, lacks ‘named entities’ in CS domain. Named entities are pre-defined categories of person, organisation, and location
	Synonyms	Synonyms hardly exists in CS domain

Architecture of the Concept Map Mining Framework

The design of the CMMF is presented using an architecture diagram in Figure 4.1. The rest of the chapter is organised according to the architecture diagram. The term ‘*CourseCMap*’ is used to introduce the concept maps generated using the CMMF.

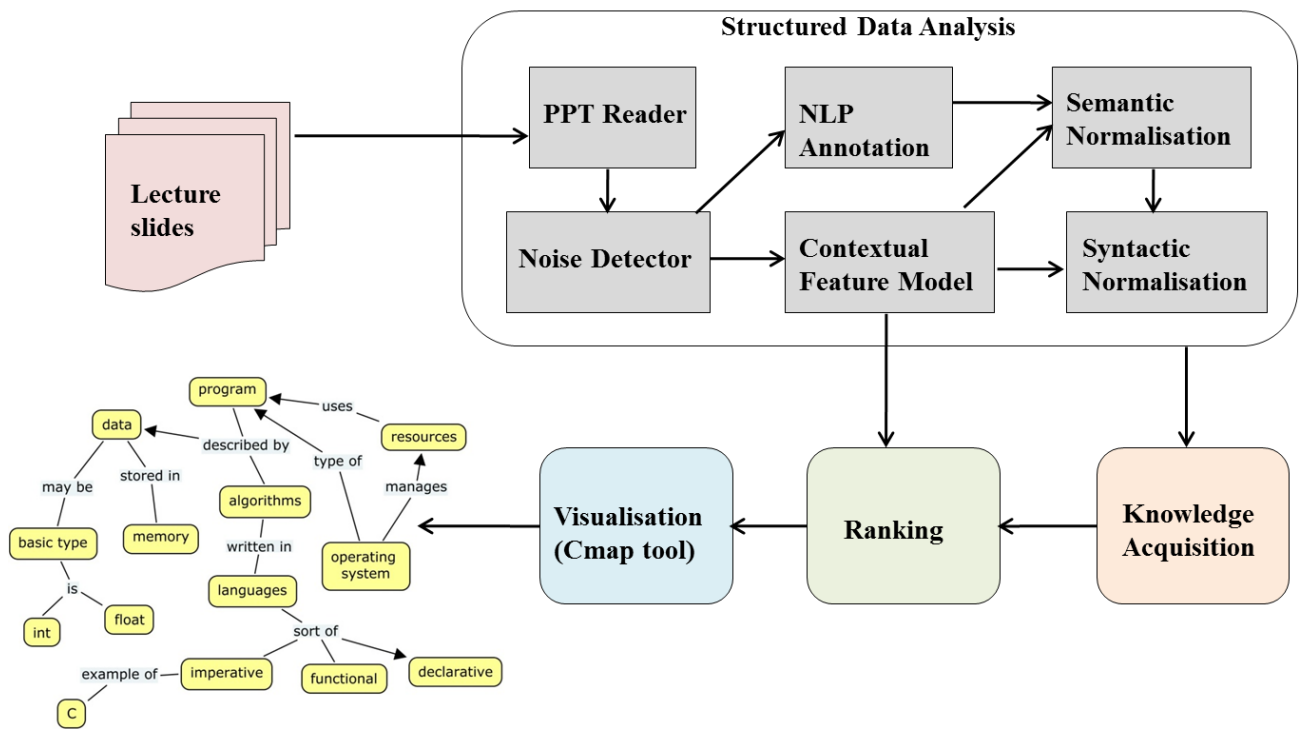


Figure 4.1: High-level architecture of the CMMF

As illustrated in the architecture diagram, the process of obtaining *CourseCMap* involved four main stages:

Structured data analysis: The aim of this stage was to analyse the source material (i.e. lecture slides) and identify and remove the element of noise.

Knowledge acquisition: This stage involved extracting useful knowledge in the form triples and arranging the triples in a hierarchy according to the natural layout of the presentation framework (Atapattu et al., 2012, 2014b).

Ranking: This stage of CMMF ranked the triples according to their importance with the use of structural features (e.g. *term frequency*) embedded in the presentation framework and graph-like representation of concept maps (e.g. *in-degree* and *out-degree*) (Atapattu, Falkner, & Falkner, 2014a).

Visualisation: The generated concept maps were visualised using IHMC CMapTools (Canas, Hill, et al., 2004) by translating them into the concept map extensible language (CXL) .

4.1.1 Structured Data Analysis: Design

The design for analysing and removing elements of noise in lecture slides followed a top-down approach to remove noise as categorised into three stages: slide-level, sentence-level and token-level.

Slide-level Noise

The slide-level noise identified slides which contain information inappropriate for inclusion in a knowledge representation such as concept maps (e.g. course announcements, exam policy and references). The hypothesis of slide-level noise detection was constructed as '*if the topic (i.e. root) of the course or title of a slide does not co-occur fully or partially (as word tokens) with the content of the slide, then the whole slide can be eliminated as unrelated*'. For instance, if the topic (i.e. *root*) of the lecture is 'software process model', related content slides might co-occur with the term 'software process model' as a whole or partially with 'software process', 'process model', or 'process' tokens. This hypothesis was based on an analysis of the corpus. The analysis found that it is infrequent to include '*course announcements*' in same slide as important subject contents. This clear separation of important and noisy data across slides simplified the process of slide-level noise detection.

Sentence-level Noise

Sentence-level noise involved syntactically and semantically incomplete, incorrect or ambiguous sentences which made it undesirable to apply linguistic techniques. Due to the point-based nature of lecture slides, the majority of sentences were grammatically incomplete. This

included verb phrases without a proper subject, or noun phrases without an object as shown in an example slide in Figure 4.2.

Development testing

- Development testing includes
 - Unit testing
 - should focus on testing the functionality of objects or methods.
 - Component testing
 - should focus on testing component interfaces.
 - System testing
 - should focus on testing component interactions.

Figure 4.2: Sample slide to illustrate incompleteness of sentences

Lecture slides often consisted of ambiguous sentences that occurred due to pronouns or determiners. A pronoun is a word that substitutes for a noun or noun phrase usually using words like *it*, *their*, *he*, *we*, *you*. Pronouns are also referred to ‘*anaphora*’ in the context of computational linguistics. The use of pronouns and determiners occurred mainly when the narration is incorporated into the presentation, particularly when the author referred to the students who use the lecture material or who are listening to the lecture (see Figure 4.3). There is a possibility of semantic information loss if pronouns are not properly replaced from the context (Zubrinic et al., 2012).

Two Dimensional Arrays

	X	
O		X
	O	

- Suppose **we** wish to represent the current state of a game of noughts and crosses
 - One way is to represent **this** by an array with nine elements
 - However **it** is difficult to keep track of what square the index corresponds to
 - This would be even more difficult if **we** were dealing with a larger grid
 - Imagine a 8x8 grid used in chess or a 9x9 grid used in Sudoku

Figure 4.3: Sample slide to illustrate semantically ambiguous sentences

The widely used approach for pronoun resolution relied heavily on linguistics and domain knowledge and generally utilised ‘named entities’ to solve gender-related ambiguities (Leskovec, Grobelnik, & Milic-Frayling, 2004). Leskovec et al. (2004) assumed that pronouns

in the text could refer only to ‘named entities’ mentioned in the same text and resolved five basic types of pronouns and their different forms: ‘he’ (*himself, his, him*), ‘she’, ‘I’, ‘they’ and ‘who’. The performance results are presented in Table 4.2.

Table 4.2: Performance of pronoun resolution by Leskovec et al. (2004)

Pronoun	Frequency	Accuracy (%)
He	681	87
They	244	67
I	64	83
She	24	62
Who	11	64
Total		82

Mitkov (1998) introduced an approach independent from ‘named entities’ which was based on searching for replacement candidates in the same sentence or backward and forward search of preceding and succeeding sentences. In his work, Mitkov (1998) adopted part-of-speech tagging and a simple noun phrase rule to determine replacement candidates. This rule identified noun phrases which precede the pronoun within a distance of two sentences, gender and number agreement (i.e. singular or plural). This approach achieved an overall accuracy rate of 90% for *technical manuals*. However, some of the indicators mentioned in his work considered a pre-defined word list. According to Mitkov’s (1998) heuristics, “if the first noun phrase of the previous sentence is followed by one of the list of defined verbs (known as *indicating verbs*) such as *discuss, present, illustrate, identify, check, develop*, then that noun phrase has a higher chance of being selected as a replacement candidate”.

The issues related to ‘named entities’ in CS domain and the lack of logical relationships between preceding or succeeding bullet points (see Table 4.1) leads to the development of a new approach for pronoun resolution within lecture slides.

Similarly, there is no specific method to resolve determiners (see Table 4.3 for examples found in the corpus) commonly occurring in lecture slides. A determiner is a word or phrase that occurs together with a noun or noun phrase which expresses the reference of that noun or noun phrase in the context (e.g. *these, this*). Previous work in resolving determiners prompted the users to choose the corresponding replacement (e.g. *Cloud* (Alves et al., 2001)). Intermediate human involvement reduced the level of automation, while increasing accuracy. This leads to the development of new techniques to replace determiners within this thesis.

Table 4.3: Examples of demonstrative determiners

Course	Topic	Determiner	Replacement
Software Engineering	Software process models	these models	Software process models
Operating System	System calls	these calls	System calls
Artificial Intelligence	Goal-based agents	this agent	Goal-based agent

In order to overcome syntactic and semantic-level issues, this thesis utilises the contextual features embedded in lecture slides. The contextual feature model has emerged based on the ‘word window model’ proposed by Ide & Veronis (1998).

Contextual Feature Model

The ‘word window model’ is a valid approach for an open problem in linguistics, called ‘word sense disambiguation’ (Ide & Veronis, 1998). It considered a window of n words to the left and right of the ‘ambiguous term’ to determine the context of the target word. The context was used to find a suitable replacement candidate for the ambiguous term or a sentence. The window can be selected as several words in same sentence, several sentences in a paragraph, slide or a document. Based on the idea of the ‘word window model’, this research utilised contextual information embedded in lecture slides to resolve ambiguity in sentences or fragments. To support the claim, this research assumes that the ‘*slide heading reflects the content in that particular slide*’ and further, ‘*each sentence (or bullet-point) shares logical relations with its enumerated sub points*’.

This implies that successive indentation levels further elaborate the information presented in the current level. However, there is no guarantee that there is any existence of logical relations between preceding and succeeding bullet points in the same indentation level. This idea is clearly illustrated in Figure 4.4 using four indentation levels (i.e. location of the text) widely used in presentation frameworks.

Figure 4.4 does not purposely mention the direction of the relation ‘*provides_context_for*’. The reason for this is that obtaining context may be done in both directions based on the application. For instance, the pronoun resolution looks backward for ‘parent’ context, while finding substitutes for sentence ‘object’ looks forward for ‘child’ context.

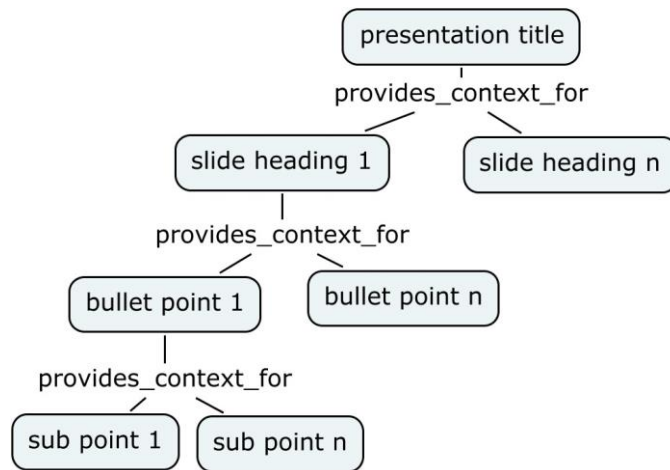


Figure 4.4: Contextual features embedded in presentation framework

Token-level Noise

The general procedure of token-level noise elimination includes processing and replacing non-alphanumeric symbols and removing *stop words* (Manning et al., 2008). However, the knowledge acquisition of CMMF completely relied on the grammatical structure of the text which only considered nouns, verbs, and their compound patterns along with adjectives and adverbs (see Table 4.11). Therefore, CMMF disregarded the typical stop word removal process used in information retrieval (Manning et al., 2008).

4.1.2 Knowledge Acquisition: Design

According to the CMM definition by Villalon & Calvo (2008), the CMM process can be expressed as the proper extraction of a concept map from a document. This process has four steps; concept extraction, relation extraction, hierarchy extraction, and ranking. The knowledge acquisition which is the core of CMMF included concept, relation, and hierarchy extraction.

In the concept mapping perspective, a *concept* is defined as a perceived regularity or pattern in events or objects, or records of events or objects, designated by a label (Ausubel, 1968, 2000; Novak & Gowin, 1984). When a concept is considered as an object, ‘words’ are used as a way to label concepts (e.g. *bird, features*) (see Figure 1.1). Flavel et al. (2002) defined a concept as a mental grouping of different entities into a single category on the basis of some underlying similarities.

When this idea is mapped into the intelligence of machines (i.e. Artificial Intelligence), concept identification is the task of finding the relevant terminology (also known as *terms* or *key terms*) in a domain after eliminating synonyms which share the common meaning (e.g. *student* and *learner*) (Cimiano, 2006; Villalon & Calvo, 2009). Previous works in the area of relevant terminology identification included information retrieval, linguistic and statistical based

methods. The works that identified synonyms with the use of external databases like WordNet (Miller et al., 1990) were applied to more *general domains* such as concept extraction from essays written about ‘English as a global language’ (Villalon & Calvo, 2009). The lecture slide sets investigated in this thesis found that synonyms hardly exist in a domain like Computer Science (Table 4.1), and if they do exist, these synonyms are not defined in lexical databases like WordNet. This led to identify Computer Science terminology using NLP techniques, and filter them to identify only the relevant terminology using statistical methods. The relevant terminology can be used as a basis to form concepts in the Computer Science domain.

Due to the issues presented in Table 4.1, it is difficult to reuse previously developed techniques for relation extraction (see the review of ‘relation extraction’ in Section 3.2). Instead, CMMF utilised the grammatical structures of text to decompose knowledge as fragments in the form of concepts and relation to identify triples. ‘Triple’ in the thesis context is used to introduce an expression or a meaningful statement either stated as subject-verb-object (SVO) or noun-verb-noun to form concept-relation-concept (Novak & Gowin, 1984). The idea of triple was originally defined in the RDF (Resource Description Framework) framework by Lassila & Swick (1999). RDF is based upon the idea of making statements about resources (web resources within RDF) in the form of subject-verb-object expressions, known as ‘triples’.

The design of knowledge acquisition from lecture slides considered a full-scale and fine-grained approach to extract all the important information without any information loss. Therefore, knowledge acquisition followed the procedures listed below.

1. Parse the text using linguistic parsers and obtain the grammar tree.
2. Identify whether the text is a simple or complex sentence or a fragment using the grammar trees produced in Step 1. A simple sentence has a clear separation of subject, verb and object while complex sentences include indirect objects, dependant clauses and nested sentences.
3. If a complex sentence is identified, decompose it into simple sentences.
4. If the sentence is simple, extract subject-verb-object (SVO) triples.
5. Extract noun-verb-noun combinations from the remaining sentences and apply filters to identify concept-relation-concept triples.
6. Extract important concepts from remaining text to define hierarchical relations.

The triples identified in step 4 and 5 produced non-taxonomic relations that occur at sentence-level. An example is shown in Figure 4.5 (*Interface testing detects faults due to interface errors*).

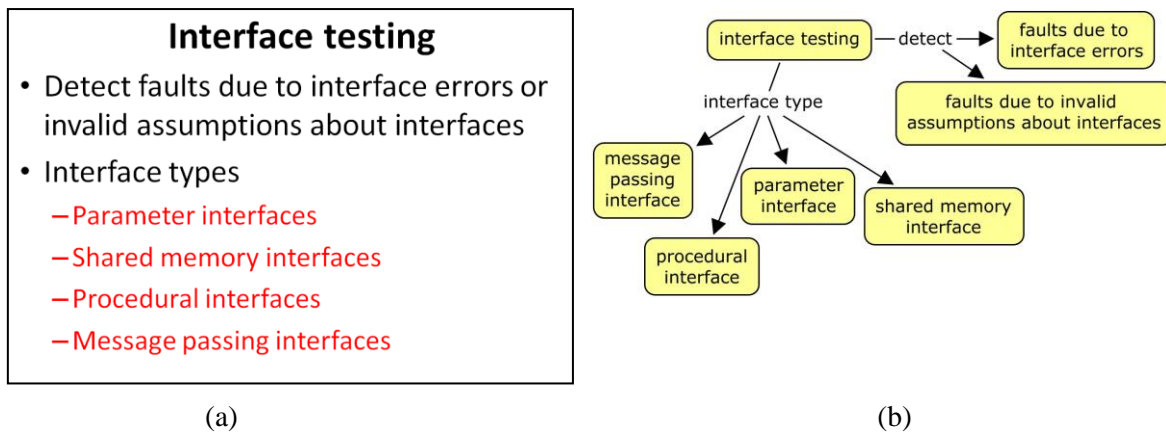


Figure 4.5: (a) Sample lecture slide (b) corresponding concept map generated using CMMF

In contrast to non-taxonomic relations, there are other types of relations called *hierarchical relations* (also known as *taxonomic relations*). As reviewed in Chapter 2, hierarchical relations are usually discovered by special text patterns called *lexico-syntactic* patterns (e.g. *such as, and other, including*) (Hearst, 1992). They create generalisation/specialisation (*is-a*) relations to identify parent-child (see Figure 3.2). As presented in Table 4.1, these types of patterns rarely exist in lecture slides.

Thus far, the focus of the thesis was on the lecture slides as a sequence of text. In fact, lecture slides often have a layout or structure which includes useful information (Rosenfeld, Feldman, & Aumann, 2002). Therefore, this research introduced a new approach to identify more general and more specific concepts according to the natural layout of the presentation framework, assuming ‘*concepts included in lecture slides are further elaborated using the next level(s) of indentation*’. This suggested that important parent concepts tend to end up as titles, with subordinate and related information on the same slide as smaller bullet points. The hierarchy extraction process utilised contextual features explained in Section 4.1.1 (see Figure 4.4). An example of hierarchy identification is shown in Figure 4.5, defining the general concept as *interface testing*, while specific concepts as *parameter interface*, *shared memory interface*, *procedural interface* and *message passing interface*.

4.1.3 Ranking: Design

The final stage of CMM is ranking (or summarisation) of the extracted knowledge according to their importance (Villalon & Calvo, 2008). According to Novak & Gowin (1984), generally, a concept map should form an overview with approximately 15 to 25 concepts, organising the *most important knowledge* in a domain or a *focus question*.

However, in the educational context, particularly in course materials, the majority of the knowledge presented is important to the learner, resulting in a large portion of lectures or textbooks being retrieved and identified for knowledge organisation. Therefore, some questions arise to determine ‘*what are the most important information in education materials?*’, ‘*how to define the most important knowledge in education materials?*’ and ‘*who is defining the importance of knowledge in education materials?*’. In order to answer above questions, this thesis contributes the design of a methodology to assess the importance of concepts extracted from lecture slides (Atapattu et al., 2014b). Three ranking models (*baseline*, *linguistic* and *structural*) were introduced to rank the concepts according to their importance. These ranking methods were evaluated using the correlation with manual ranking as performed by lecturers, as experts within the domain.

The baseline model utilised the ‘concept location’ determined by the presentation framework to decide the importance of concepts. This model hypothesised that ‘*text location allocated by the natural layout of the presentation slides might influence human judgements of which concepts are most important*’.

The linguistic model hypothesised that the ‘grammatical structure’ of text can be utilised to determine the importance of concept as ‘*simple grammatical structures (nouns, noun phrases) of lecture slides might have higher influence than complex grammatical structures (nested sentences, dependent clauses, indirect objects) on human judgement of which concepts are most important*’.

The structural feature model hypothesised that the ‘*structural (e.g. term frequency) and graph-based features (e.g. proximity) might influence the human judgement of which concepts are most important*’.

The structural and graph-based features are further explained using examples in Figure 4.6 to 4.8.

Term frequency

The frequently occurring terms are emphasised as a ‘word cloud’. For instance, the word *program* occurs 7 times and *execution* is repeated 3 times in the example slide (Figure 4.6).

Operating System Services

- Operating systems provide an environment for **execution** of **programs** and services to **programs** and **users**
- One set of **operating-system services** provides functions that are helpful to the **user**:
 - **User interface** - Almost all **operating systems** have a **user interface** (UI).
 - Varies between Command-Line (CLI), Graphics User Interface (GUI), Batch
 - **Program execution** - The system must be able to load a **program** into memory and to run that **program**, end **execution**, either normally or abnormally (indicating error)
 - **I/O** operations - A running **program** may require **I/O**, which may involve a **file** or an **I/O** device
 - **File-system** manipulation - The **file system** is of particular interest. **Programs** need to read and write **files** and directories, create and delete them, search them, list **file** information, permission management.

Operating System Concepts - 8th Edition 2.1 Silberschatz, Galvin and Gagne ©2009

Figure 4.6: Term frequency using ‘Operating System’ slide

Degree of co-occurrence

When two terms are co-occurring frequently, it is assumed that they have a strong relationship (Manning et al., 2008). In the Figure 4.7, the two terms *program* and *execution* co-occur repeatedly.

Operating System Services

- Operating systems provide an environment for **execution of programs** and services to programs and users
- One set of operating-system services provides functions that are helpful to the user:
 - User interface - Almost all operating systems have a user interface (UI).
 - Varies between Command-Line (CLI), Graphics User Interface (GUI), Batch
 - **Program execution** - The system must be able to load a program into memory and to run that **program**, end **execution**, either normally or abnormally (indicating error)
 - I/O operations - A running program may require I/O, which may involve a file or an I/O device
 - File-system manipulation - The file system is of particular interest. Programs need to read and write files and directories, create and delete them, search them, list file information, permission management.

Operating System Concepts - 8th Edition 2.1 Silberschatz, Galvin and Gagne ©2009

Figure 4.7: Degree of co-occurrence using ‘Operating System’ slide

Degree centrality

As discussed in Section 3.3, *in-degree* is defined as a count of the links directed to the node and *out-degree* is defined as the number of links that the node directs to others (Diestel, 2005). When a concept has greater out-degree than in-degree, that concept is considered as a more general concept (e.g. out-degree of *software testing* is 3 in the Figure 4.8) while the latter is identified as more specific concept (e.g. in-degree of *unit testing* is 1 in the Figure 4.8).

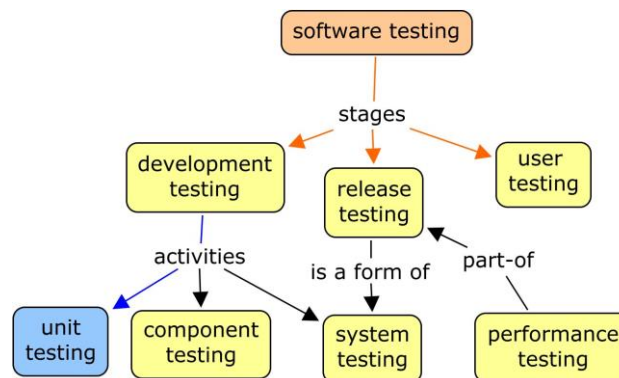


Figure 4.8: An illustration of degree centrality

Typographic information

Lecture slides often emphasise important concepts using typographic features including different font colour, font size, underlining and the use of italics. This can be utilised as one of the factors to determine the importance of concepts extracted from lecture slides. For instance, Figure 4.9 illustrates the emphasised terms such as *asymmetric clustering* and *parallelization* using a different font colour.

Clustered Systems

- Like multiprocessor systems, but multiple systems working together
 - Usually sharing storage via a **storage-area network (SAN)**
 - Provides a high-availability service which survives failures
 - ▶ **Asymmetric clustering** has one machine in hot-standby mode
 - ▶ **Symmetric clustering** has multiple nodes running applications, monitoring each other
 - Some clusters are for **high-performance computing (HPC)**
 - ▶ Applications must be written to use **parallelization**

Figure 4.9: Typographic information in a sample lecture slide

Proximity

The distance from the *root* concept to a particular concept is considered as proximity (inclusive). It was hypothesised that concepts that have a closer proximity to the *root* are expected to be more important than those with further proximity (Leake et al., 2004).

4.2 Development of the Concept Map Mining Framework

4.2.1 Development Environment

The CMMF was written using the Java language (version 1.7). The text extraction from PowerPoint utilised Apache POI API (Apache, 2011), a Java API for Microsoft documents. It supports the extraction of structural information from text such as text location (e.g. *title*, *bullet offset*) and rich text features (e.g. *font colour*, *font size*, and *underline*).

CMMF adopted Stanford Core NLP tools for the task of triple extraction from *simple sentences* (Klein & Manning, 2003a, 2003b; Manning et al., 2014). Stanford Core NLP library is commonly used as a research tool. The Stanford part-of-speech tagger is accurate up to 97% to identify part-of-speech tags of natural English texts (Toutanova, Klein, Manning, & Singer, 2003). The link grammar parser was utilised to process *complex sentences* (Sleator & Temperley, 1991). These tools form the fundamental base upon which the new techniques and algorithms described in this thesis were constructed.

The extracted triples were stored using a XML-based file format called Concept Map Extensible Language (CXL) and visualised using IHMC CMapTools (Canas, Hill, et al., 2004).

4.2.2 Natural Language Annotation

This research utilised NLP tools and techniques for almost all stages of the framework including structured data analysis, knowledge acquisition and ranking. This section explains the NLP annotations: part-of-speech tagging, lemmatisation and syntactic parsing using Stanford statistical parser.

The Stanford Core NLP Project is an integrated framework of NLP tools such as part-of-speech (POS) tagger, parser, named entity recogniser, sentimental analysis, and lemmatiser (Manning et al., 2014). The following Java code illustrates the integration of Stanford Core NLP tool to CMMF.

Java implementation of NLP Annotation

```
Properties properties = new Properties()
properties.put("annotators","tokenize,ssplit,pos,lemma,parse") //annotators
StanfordCoreNLP pipeline = new StanfordCoreNLP(properties)

Annotation annotation = new Annotation(text) //text - input text
pipeline.annotate(annotation)

List<CoreMap> sentences = annotation.get(SentencesAnnotation.class)

foreach sentence in sentences as CoreMap
    Tree tree = sentence.get(TreeAnnotation.class) //tree annotation

    foreach token sentence.get(TokensAnnotation.class) as CoreLabel
        String pos = token.get(PartofSpeechAnnotation.class) // pos
        String lemma = token.get(LemmaAnnotation.class) //lemma

    end foreach
end foreach
```

Lemmatisation

Lemmatisation techniques were applied to map the words into their base form to avoid multiple concepts in the same base form (e.g. *activities* => *activity*, *normalisation* => *normalise*). This research has selected lemmatisation over the popular *stemming* (Porter, 2006) or *morphological analysis* techniques since the latter two techniques will entirely remove suffixes of the term. This often gives a different meaning for the term (e.g. *computation*, *computing*, *computer* => *compute*).

Part-of-speech tagging

The assignment of part of speech in word-level such as noun, verb, and adjective is substantial to identify the grammatical structures of input text. Table 4.4 summarises most commonly used part-of-speech tags within this thesis (Marcus et al., 1994).

Table 4.4: Part-of-speech tags (word-level) used in Penn Treebank project

Tag	Description
CC	Coordinating conjunction (e.g. <i>and</i> , <i>or</i>)
DT	Determiner (e.g. <i>the</i> , <i>a</i>)
IN	Preposition or subordinating conjunction (e.g. <i>of</i>)
JJ	Adjective
NN	Noun (singular)
NNS	Noun (plural)
PRP	Personal Pronoun (e.g. <i>he</i> , <i>it</i>)
PRP\$	Possessive pronoun (e.g. <i>its</i> , <i>his</i>)
RB	Adverb
VB	Verb
VBG	Verb, gerund or present participle

The NLP annotations of part-of-speech tagging and lemmatisation are further illustrated using an example sentence ‘Waterfall model uses stable requirements in larger systems’ in Table 4.5.

Table 4.5: NLP Annotations of an example sentence

Word	Waterfall	model	uses	stable	requirements	in	larger	systems
Lemma	Waterfall	model	use	stable	requirement	in	larger	system
POS	NN	NN	VBZ	JJ	NNS	IN	JJR	NNS

Parsing

The parse tree (phrase structure grammar) of input sentence simplified the identification of subject-verb-object and dependencies. Figure 4.10 illustrates the parse tree of the example sentence ‘Waterfall model uses stable requirements in larger systems’ using the Stanford statistical parser (Klein & Manning, 2003a, 2003b).

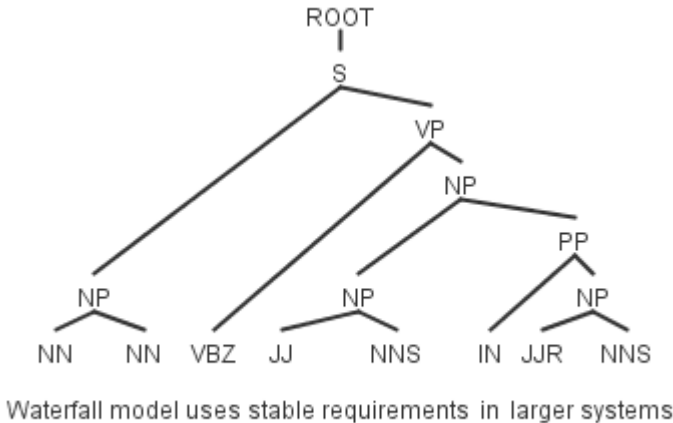


Figure 4.10: Parse tree of an example sentence

CMMF utilised these existing tools as a basis upon which to define new algorithms and techniques, specific to the context and to address the issues associated with lecture slides. This thesis proposed new techniques for noise detection, pronoun resolution, triple extraction, hierarchy extraction, and ranking.

4.2.3 Structured Data Analysis

This section discusses the text analysis used to identify semantically ambiguous and syntactically incomplete text.

PowerPoint Reader

The initial phase of ‘structured data analysis’ was parsing the document through a *PowerPoint Reader* (Apache, 2011). PowerPoint Reader keeps track of structural information of text such as text location (e.g. title, bullet point, bullet offset) and rich text features (e.g. font colour, font size, underline) to assist ranking the extracted concepts. The structure of POI objects and corresponding methods are illustrated in Figure 4.11.

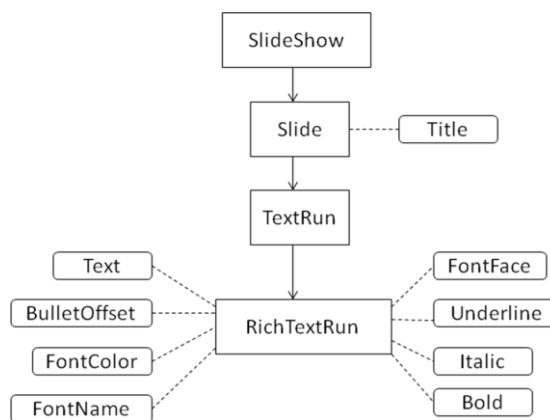


Figure 4.11: The structure of the Apache POI for Microsoft PowerPoint documents

Sentence boundary detection

The document parsed through the PowerPoint Reader usually consists of bullet points. The first step was to split bullet points into sentences since CMMF processed textual data as sentences. This process used regular patterns to detect sentence boundaries. If there was a period (e.g. *full stop, question mark*) in between sentences and the following character was a space or a capital letter followed by a space, the algorithm detected such text as start of a new sentence. This approach distinguished other characters which use periods within the sentence such as *e.g.* and *i.e.*

Regular expression for sentence boundary detection: $\backslash \cdot \backslash s [A-Z] | \backslash \cdot [A-Z] | \backslash ? \backslash s [A-Z]$

Slide-level Noise

As discussed in the design of the CMMF in Section 4.1, lecture slides often contain information inappropriate for inclusion in a knowledge representation such as a concept map (e.g. *course announcements, references, assessment details*). Therefore, based on the developed hypothesis ‘if the topic (i.e. root) of the course or title of a slide does not co-occur fully or partially (as word tokens) with the content of the slide, then the whole slide can be eliminated as unrelated’, the co-occurrence between titles (topics and slide titles) and other text content was calculated. Prior to applying the automated noise elimination algorithm on slide data, footers from all slides were removed automatically since footers sometimes contain the topic of the lecture which can affect to the underlying technique of noise detection. Slide titles might include category headings such as *introduction, Chapter 1*, etc.. In such occasions, the content of those particular slides should fully or partially co-occur with the ‘topic’ of the lecture in order to consider the particular slide as important.

The analysis of co-occurrence between titles and body text was performed using both *greedy* and *non-greedy* approaches. The *greedy approach* starts with the largest ‘text pattern’, and if matches are found in the input text, extracts, then, applying the next largest ‘text pattern’ to the remaining text recursively until no any other ‘text patterns’ are found in the text. The commonly used text patterns in this thesis are presented in Table 4.11 (page 82).

In contrast, the non-greedy approach applies all of the ‘text patterns’ into the full text, instead of considering only the remaining text which is the case in the greedy approach.

Table 4.6 illustrates the features considered to analyse co-occurrence for automated noise detection in slide-levels.

Table 4.6: Features of co-occurrence calculation for noise detection in slide-level

	Root (Topic)	Tokens of root	Slide title	Tokens of Slide title	Body text (greedy)	Body text (non-greedy)
Root (Topic)	-	-	✓	✓	✓	✓
Tokens of root	-	-	✓	✓	✓	✓
Slide title	✓	✓	-	-	✓	✓
Tokens of slide title	✓	✓	-	-	✓	✓
Body text (greedy)	✓	✓	✓	✓	-	-
Body text (non-greedy)	✓	✓	✓	✓	-	-

In this approach, only nouns or noun phrases followed by adjectives were considered as ‘tokens’ (see Table 4.11). This prevented unimportant tokens such as stop words (e.g. *the, therefore*) to be considered for co-occurrence calculation. The elimination of slide-level noise beforehand

reduced the computations. Yet, useful knowledge extraction from lecture slides is still constrained due to noisy data occurring at sentence-level.

Sentence-level Noise: Semantic Normalisation

Pronoun Resolution

It is likely that useful information is lost when pronouns are included in text and these pronouns are not replaced properly using the referring terms. As presented in Table 4.1, due to various issues associated with CS domain and lecture slides, previously developed pronoun resolution algorithms cannot be reused.

Initially, backward search was performed to find replacement candidates in the sentence itself, and in the preceding and succeeding sentences if the bullet-point contains multiple sentences (Mitkov, 1998). In addition, CMMF found replacements in ‘parent-levels’ which is the preceding indentation level or title of the slide (see parent context for each element in Figure 4.4). The developed heuristics were limited within the context of a single slide. This is due to the fact that there was no evidence found through corpus analysis that the context of a pronoun is included across multiple slides. It is difficult to navigate between slides to find the context of a pronoun not only for machine algorithms but also for humans.

Table 4.7 illustrates the features considered for pronoun resolution. Some of these features are specific to presentation framework such as location, sub-location while other features obtained from the work by Mitkov (1998) (e.g. grammatical number).

Table 4.7: Features for pronoun resolution

Feature	Description
Location	Same sentence Preceding sentence in same bullet-point Parent indentation-level or title
Sub-location	Head word None head word
Distance	Distance from pronoun to candidate replacements using token count
Grammatical structure	Grammatical structure of candidate replacements such as noun, noun phrase or adjective
Grammatical number	Singular or plural
Type of pronoun	Personal or possessive pronoun
Insolvable pronouns	<i>We, you, us, itself</i>

Algorithm 4.1 explains the steps for resolving a single pronoun using the context of a single slide. When an input sentence contained more than one pronoun, the pronouns were resolved by

starting from the beginning of the sentence, replacing the first pronoun and recursively working on the other pronouns until the full sentence becomes unambiguous.

Algorithm 4.1: Pronoun resolution

```
Require: pronoun      // pronoun to resolve
           precede text // text precede the pronoun
           parent       // parent text of pronoun

1.Candidates <- find candidates in the same sentence U
2.if number of sentences per bullet point > 1 or candidates = 0
   candidates <- find in precedence sentence
   else if number of sentences per bullet = 1 or candidates = 0
     candidates <- find in parent
3.end if

4.if candidates > 1
   4a if possessive pronoun (PRP$)
     i. if followed by punctuations or includes in SBAR
        replacement <- nearest precede term to pronoun or
           otherwise in matching grammatical number
     ii. end if
   4b end if

5.else
   replacement <- find in headword or otherwise in matching grammatical
   number
6. end if

Output: replacement      // resolved pronoun
```

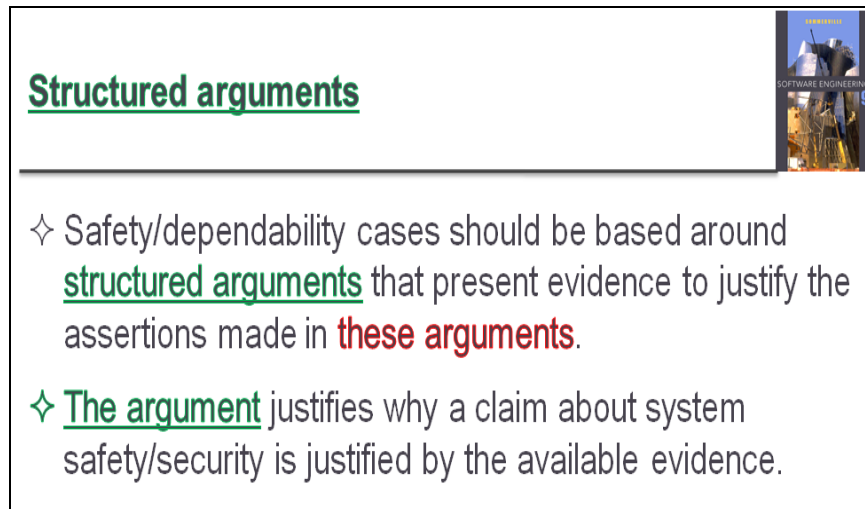
Demonstrative Determiners

Due to the typical excessive use of informal language, lecture slides often contain demonstrative determiners (e.g. *this*, *these*, *those*, *that*), which have not been resolved previously by existing pronoun resolution algorithms. Alves et al. (2001) proposed an approach to resolve determiners with the involvement of user inputs.

Unlike pronouns that have a specific part-of-speech tag (PRP or PRP\$ - see Table 4.4), demonstrative determiners do not have a dedicated tag. They share the tag ‘DT’ with other definite and indefinite articles (e.g. *the*, *a*) and possessive determiners (e.g. *my*, *their*). Therefore, identification of a demonstrative determiner using grammatical annotations is challenging. CMMF focused on resolving four determiners - *this*, *these*, *those*, *that*.

The solution presented here primarily considered lexical reiterations (repetitions of lexical item - see Table 4.3 for an example). Additionally, features like grammatical number (either singular or plural), distance from the determiner to the probable candidates (using token count), number of strings overlapping with the candidate, and the grammatical structure between candidates and demonstrative determiners were considered. For instance, if the ambiguous term contained part-

of-speech pattern [DT NNS], the candidate pattern should be something like ‘NN / NNS / NNP followed by NNS’, stressing the inclusion of NNS as the succeeding tag.



Structured arguments

- ✧ Safety/dependability cases should be based around **structured arguments** that present evidence to justify the assertions made in **these arguments**.
- ✧ **The argument** justifies why a claim about system safety/security is justified by the available evidence.

Figure 4.12: An example slide with demonstrative determiners

For instance, in the example of Figure 4.12, candidate replacements of ‘*these arguments*’ are ‘*structured arguments*¹’ in the title, ‘*structured arguments*²’ in the same sentence and ‘*the argument*’ in next bullet-point. Based on the grammatical number of the determiner (i.e. plural), the candidate ‘*the argument*’ is eliminated. From the remaining candidates, ‘*structured arguments*²’ in the same sentence will be considered as the replacement since the distance from determiner to this candidate is low (9 token counts) compared with the other candidate (i.e. *structured arguments*¹) resides in the title (17 token counts).

Sentence-level Noise: Syntactic Normalisation

In addition to the semantic ambiguities occurring at the sentence-level, the majority of the sentences included in the lecture slides were grammatically incomplete by nature. This occurred due to the point-based nature of the presentation framework. The presentation framework often encourages writing of informal sentences, fragments or colloquialisms (known as *bullet points*). This kind of text restricts the application of existing NLP-based algorithms to identify useful knowledge. Therefore, this section explains the application of the contextual features (see Figure 4.4) to resolve syntactic-level ambiguities.

Subject or object allocation

In order to overcome the issue of sentence fragments, CMMF introduced an approach which nominates syntactically missing elements. Syntactic parse trees of sentence fragments were obtained using the Stanford parser (Klein & Manning, 2003a, 2003b) (see Figure 4.13 and 4.14).

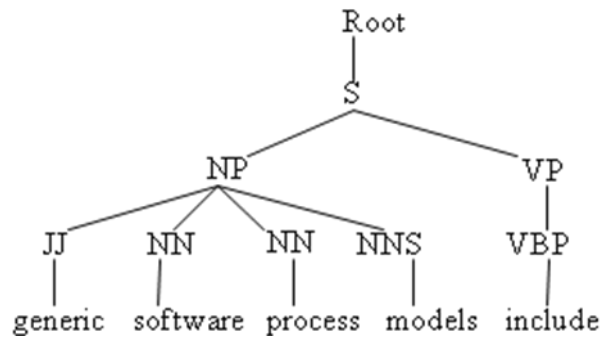


Figure 4.13: A parse tree of an example NP ‘Generic software process models include’

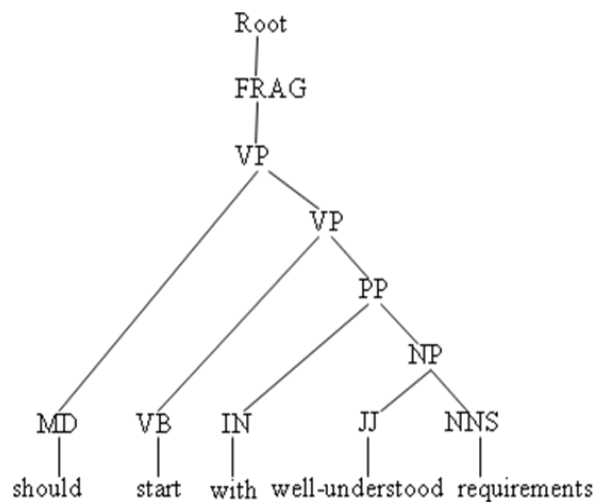


Figure 4.14: Parse tree of an example VP ‘should start with well-understood requirements’

In addition to the Penn Treebank tags discussed using word-level (see Table 4.4); there are other types of tags used to identify phrase- and clause-levels as shown in Table 4.8 (Marcus et al., 1994).

Table 4.8: Phrase- and clause-level Penn TreeBank tags

Level	Tag	Description
Phrase	ADJP	Adjective phrase
	ADVP	Adverb phrase
	FRAG	Fragment
	NP	Noun phrase
	PP	Prepositional phrase
	VP	Verb phrase
Clause	S	Simple declarative clause
	SBAR	Clause introduced by a subordinating conjunction
	SBARQ	Direct question introduced by <i>wh</i> -word or <i>wh</i> -phrase
	SINV	Inverted declarative sentence
	SQ	Inverted yes/no question

Based on the corpus analysis, two main types of phrases used in lecture slides were noun phrases (NP) and verb phrases (VP). Noun phrases generally contain one or more nouns followed by a verb (Figure 4.13). Therefore, noun phrases require an ‘object’ to create a subject-verb-object combination in a sentence. Similarly, verb phrases contain a verb followed by one or more nouns (Figure 4.14). This structure requires a ‘subject’ to form a complete sentence.

The idea of ‘subject’ or ‘object’ allocation is explained with the use of Figure 4.15 below.

Generic software process models

- Most software process models are based on one of the three generic models
- Generic software process models include NP
 - 1 Waterfall model
 - 2 Should start with well-understood requirements VP
 - Incremental development
 - 3 Identifies and resolves technical risk early
 - Re-use oriented development
 - Reduces the cost and development time

Figure 4.15: Sample lecture slide from Software Engineering course

According to the Figure 4.15,

- Noun phrases which contain the pattern [NP VP [VB]] (Figure 4.13) look forward for candidates in ‘child’ levels to replace the missing ‘object’ (numbered ‘1’ in Figure 4.15). Here, ‘child’ levels imply sub-indentation levels or enumerated sub points.
- Verb phrases which contain the pattern [VP [VB NP]] (Figure 4.14) look backward for candidates in ‘parent’ levels to replace the missing ‘subject’ (numbered ‘2’ in Figure 4.15). Here, ‘parent’ levels imply preceding-indentation levels.

Candidates are nouns or compound nouns regarded as suitable for, or likely to become, the replacement of the missing ‘subject’ or ‘object’ in this context. The candidates found in the above steps were assigned with scores based on their features such as their grammatical structure (e.g. nouns, verbs), distance from ‘input fragment’ to the candidate (using number of tokens), number of tokens in the candidate, whether it is an immediate indentation level or not (either backward or forward). The assigned scores for each feature were summed to obtain an aggregate score for each candidate. The highest score candidate will be used as the replacement

of missing ‘subject’ or ‘object’. This approach assisted to transform the majority of fragments into meaningful, complete sentences. The generated sample sentences from the scenarios in Figure 4.15 are listed below.

Object allocation: *Generic software process models include waterfall model*

Subject allocation: *Waterfall model should start with well-understood requirements*

Coordinating Conjunction (CC)

In linguistics, conjunctions join two elements which include words, phrases, clauses or sentences. CMMF split verb phrases which were connected through conjunctions ‘[VP [VP CC [VP NP]]]’ and assigned ‘subjects’ to distinct verb phrases. Further, it also resolved some complex substitutions like allocating both ‘subjects’ and ‘objects’. The example shown in Figure 4.15 (numbered ‘3’) produces the following two sentences (Appendix B includes the algorithm for this).

Incremental development identifies technical risk early

Incremental development resolves technical risk early

4.2.4 Knowledge Acquisition

In automated knowledge acquisition, knowledge is primarily decomposed as fragments in the form of entities (or concepts) and relations known as ‘triples’. The organisation of these triples as hierarchical and prepositional network produces concept maps (Novak & Canas, 2006).

This section describes the core of the CMMF which extracts knowledge from lecture slides to generate concept maps. CMMF is capable of extracting useful knowledge from *well-written* as well as *ill-written* English text. Knowledge extraction from well-written text will be discussed under ‘subject-verb-object’ extraction, while the remaining discussion here considers the text as ill-written, since such texts do not have a proper distinction between subject, verb and objects. Therefore, noun-verb-noun triples or concepts will be extracted from ill-written texts as discussed under ‘triple extraction’ and ‘concept extraction’.

In order to assist better understanding of the process of knowledge acquisition, Figure 4.16 illustrates a ‘decision tree’. Each feature or combined set of features to obtain ‘decisions’ and ‘actions’ are drawn using an oval and a rectangle respectively.

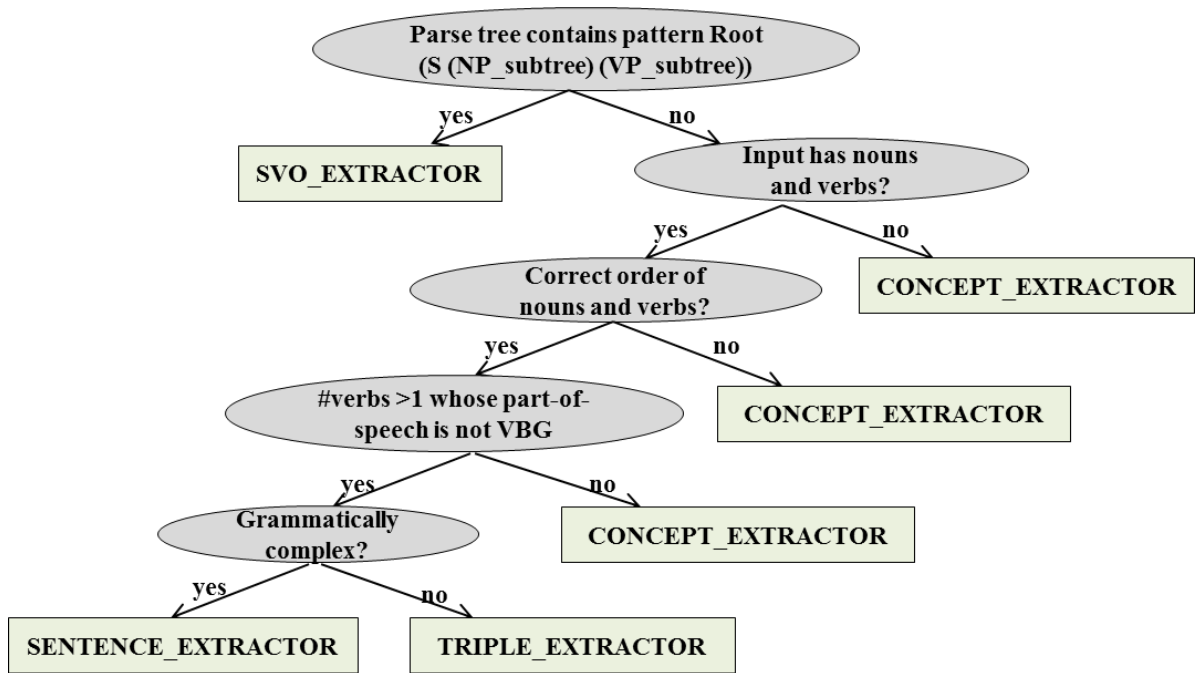


Figure 4.16: Decision tree of CMMF

There are two ‘actions’ shown in decision tree called SVO_EXTRACTOR and TRIPLE_EXTRACTOR. The earlier one refers to the apparent subject-verb-object triples included in the simple English sentences while the latter one refers to noun-verb-noun triples obtained from grammatically complex sentences and ill-written text. Therefore, the latter triples cannot be guaranteed to produce proper triples unless they are refined through a ‘likelihood filter’, which will be discussed under ‘triple extraction’.

Subject-Verb-Object (SVO)

According to the decision tree, CMMF first considered extracting subject-verb-object (SVO) triples from simple declarative sentences whose grammatical structure follows the heuristics proposed by Rusu et al. (2007) and suggested enhancements of this thesis. A simple declarative sentence is a statement whose subject normally precedes the verb with a clear distinction between subject, verb and object.

Figure 4.17 illustrates the overview of SVO extraction.

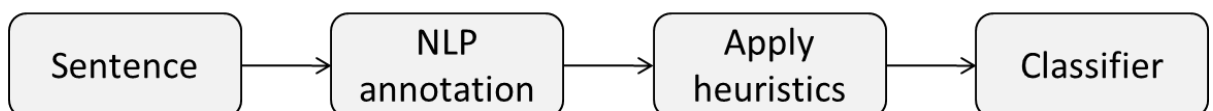


Figure 4.17: Overview of SVO extraction from simple sentences

The NLP annotation included parsing the sentences through the Stanford statistical parser (Klein & Manning, 2003a, 2003b). The application of heuristics is explained below with examples. Finally, the classifier calculated the string similarity between the SVO extracted by human and computer in order to evaluate the algorithm.

If parser tree of the input sentence (Figure 4.18) contained the pattern ‘Root (S (NP_subtree) (VP_subtree))’, it applied the following rules to extract subject-verb-object (SVO) triples.

1. **Rule 1 (subject):** Perform breadth first search in NP_sub tree and select first descendant of NP_sub tree. This can be a noun or compound nouns
2. **Rule 2 (verb):** Search in VP_sub tree for deepest verb descendent
3. **Rule 3 (object):** Search in PP, NP or ADJP siblings of the VP_sub tree. In NP or PP_sub tree, select first noun or compound noun, or in ADJP sub tree, select first adjective

Figure 4.18 highlights the subject, verb and object of an example sentence ‘*waterfall model uses stable requirements in larger systems*’.

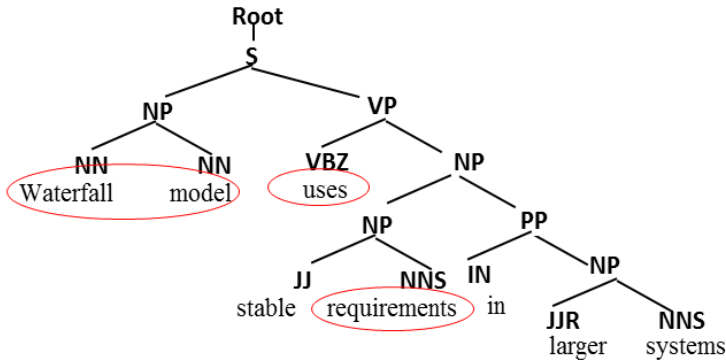


Figure 4.18: Parse tree for an example sentence with highlighted SVO

This research extended the heuristics proposed by Rusu et al. (2007) in order to improve the meanings of extracted SVO. Instead of exact nouns or compound nouns, CMMF considered extracting noun phrases that were expressed by prepositions. In some occasions, failure to extract prepositional phrases caused inability to express the intended message in the context. This idea is illustrated using an example sentence ‘*Transition phase of rational unified process deploy the system*’ in Figure 4.19.

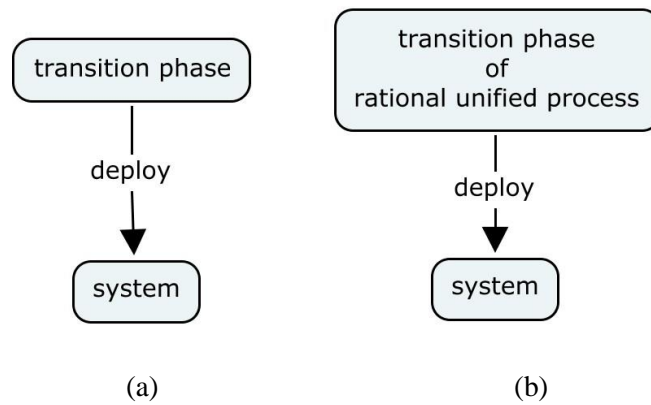


Figure 4.19: Triple extraction from an example sentence using (a) baseline algorithm (Rusu, et al., 2007) (b) CMMF

According to the decision tree, if the input sentence followed one of the patterns described above, CMMF simply extracted SVO from them (SVO_EXTRACTOR). However, there were many variations of sentence patterns found in lecture slides in addition to the ones discussed above. Even though some sentences did not possess the above patterns, they still could contain more than one subject, verb and objects in complex sentences or nouns and verbs which could act as triples. This required to introduce new algorithms to discover useful knowledge.

Complex sentence

According to the corpus under study, a sentence was defined as ‘grammatically complex’ if it had more than one nested sentences (S) preceded by dependent clauses (SBAR) and indirect objects. The parse tree of an example complex sentence ‘*In some programming languages the length of the array is fixed when it is first declared*’ (with S and SBAR highlighted) is shown in Figure 4.20.

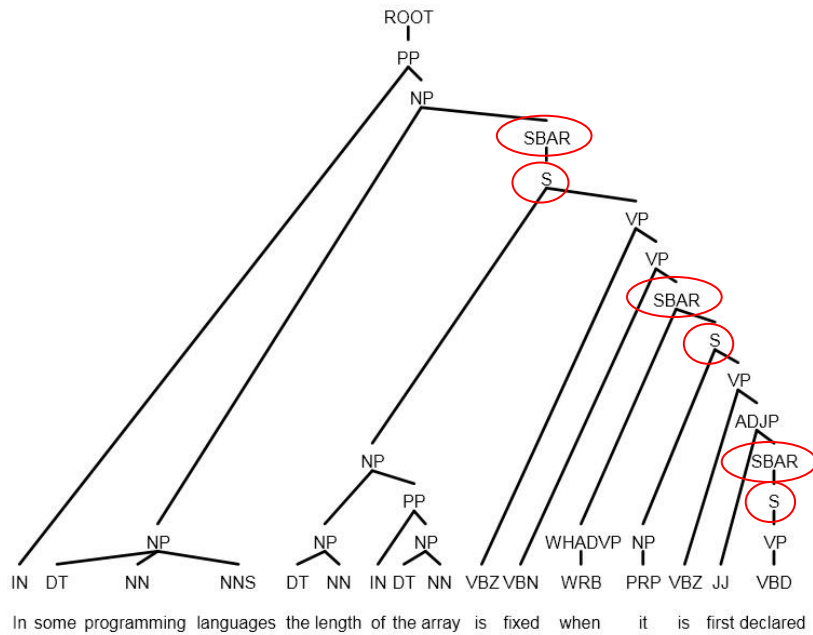


Figure 4.20: Parser tree of an example complex sentence

The complex sentence processor split complex sentences into simple sentences (SENTENCE_EXTRACTOR). The simple sentences produced at this stage might not include a clear separation between subject, verb and object in order to apply the previous heuristics (Rusu et al., 2007). Therefore, the link grammar parser was utilised to obtain triples from these split sentences (Sleator & Temperley, 1991).

Link Grammar parser

The link grammar parser is a syntactic parser for English language developed at CMU (Sleator & Temperley, 1991). It consists of labelled links which connect pairs of words. The parser includes a ‘dictionary’, which defines grammatical rules for parsing. This parser is flexible for users to define their own rules. The Table 4.9 defines list of commonly used link types in this thesis.

Table 4.9: Commonly used link types in the Link Grammar parser

Link type	Description (Connection to/from)
A	Adjectives to nouns
CC	Clauses to coordinating conjunction
D	Determiners to noun
I	Infinitive verb to modal verbs and 'to'
IN	Preposition to certain time expressions
J	Prepositions to object
M	Nouns to post-noun modifiers
Mv	Verbs and adjectives to modifying phrases such as adverbs
P	Verbs to prepositions, adjectives, passive and progressive participles
O	Verbs to objects, either direct or indirect
S	Subjects to finite verbs
TO	Verbs and adjectives to the word 'to'

Java implementation of Link grammar

```

ParseOptions parse_options = new ParseOptions()
Dictionary dictionary = new Dictionary(parse_options, "4.0.dict",
"4.0.knowledge", null, "4.0.affix")

Sentence sentence = new Sentence(text, dictionary, parse_options)

int num_of_linkages = sentence.sentence_parse(parse_options)

if(num_of_linkages > 0)
    Linkage linkage = new Linkage (0, sentence, parse_options)
    linkage.linkage_print_diagram()
end if

```

The link grammar parser usually extracts sub-linkages when it encounters complex or compound sentences. Figure 4.21 illustrates how to obtain multiple sentences from sub linkages using an example sentence ‘*Process creation is heavy-weight while thread creation is light-weight*’.

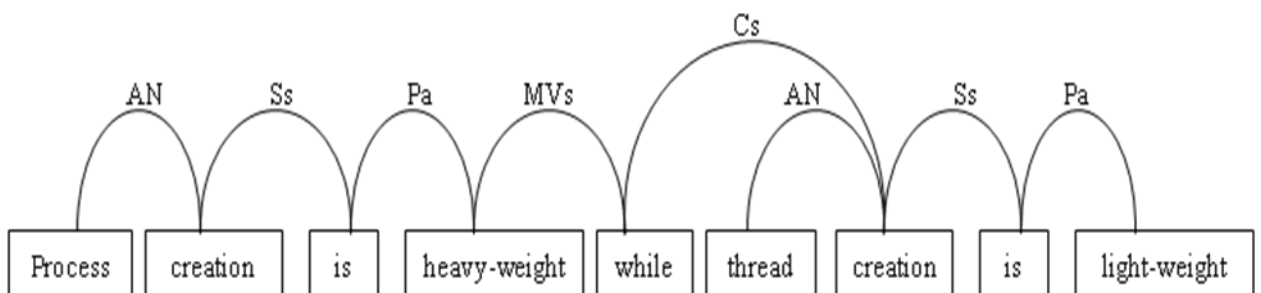


Figure 4.21: Linkage diagram of an example complex sentence

As illustrated in Figure 4.21, the two simple sentences ‘*Process creation is heavy-weight*’ and ‘*Thread creation is light-weight*’ are connected through the subordinating conjunction ‘*while*’,

denoted as MVs. The two Ss linkages denote that there are two subjects. The following heuristics are used to obtain triples from each sub linkage (Rusu et al., 2007).

1. **Rule 1(subject)**: Select the word left of S_link
2. **Rule 2(verb)**: Select first word right of S_link until [Pv, Pg, PP, I, TO, MV_i] links found
3. **Rule 3(object)**: Select links from ‘verb’ until [O, Os, Op, MV_{pn}] links found

Triple

According to the decision tree, if the input sentence contained nouns (including compound nouns) and verbs, but have no clear separation between subject-verb-object to obtain SVO triples using linguistic parsers, it is likely that those components of the text could be converted into triples. Therefore, part-of-speech of any sentences filtered out from the SVO extraction was considered to identify nouns, compound nouns, verbs along with their describing adjectives using a ‘greedy approach’. These extractions were checked against ‘order’ where a verb should be in-between two nouns to form a triple. However, the candidate list should contain at least one verb which should not be a ‘gerund verb (VBG)’ (see Table 4.1).

From those sentences, all possible combinations of noun-verb-noun triples were extracted (TRIPLE_EXTRACTOR). These triples could either be meaningful or weak. Therefore, a ‘likelihood filter’ was applied (Dunning, 1993; Olney et al., 2011). The ‘likelihood ratio’ by Olney et al. (2011) calculated whether the relation between the start and end node is meaningful. The start and end nodes were pooled into bags-of-words and the association between each of these was calculated using co-occurrence analysis. Bag-of-words model considers the tokens of sentence after removing stop words and symbols (Salton & McGill, 1986). For instance, the terms ‘program’ and ‘execution’ in *Operating system* co-occurred frequently and hence, comprised strong association (see Figure 4.7 for an example).

In addition to the ‘association’ between triples considered by Olney et al. (2011), CMMF considered some additional features to strengthen the identification of strong triples. Table 4.10 presents the most common features considered in this thesis.

Table 4.10: Features for triple extraction

Feature	Description
Number of nouns	Determine candidate ‘starting’ or ‘end’ node
Number of verbs	Determine candidate relations
Order of nouns and verbs within the sentence	Determine possibility of creating triples
Number of ‘gerund’ verbs	Determine possibility of creating connections
Subject-verb distance (using tokens)	Determine the likelihood of subject-verb relation
Verb-object distance (using tokens)	Determine the likelihood of verb-object relation
Typographic information	Determine whether candidate is emphasised using different font colour, size, face, underline, bold or italic
‘Insoluble’ pronouns	Whether sentence contain <i>we, us, you, itself</i>
Extremely complex pronouns	Determine possibility of solving pronouns using context of slide
Dummy pronouns	Pronouns whose replacement is not include in contexts

Finally, the extracted triples were checked against ‘redundancy cycles’. This means whether the subject is repeated in an object.

This research proposed a new set of features to extract triples from both well-written and ill-written English sentences. This approach is applicable regardless of the pre-defined patterns used in the work of Rusu et al. (2007). However, reusing the work of Dali & Fortuna (2008) and Rusu et al. (2007) improved the accuracy of triple extraction in specific sentence patterns. The addition of new features is a consequence of an extensive review of a large variety of Computer Science (CS) lectures undertaken as part of this thesis. This work has a potential to be reused for knowledge acquisition from any textual knowledge source written in English by eliminating features specific to the presentation framework such as presentation artefacts.

Concept

According to the decision tree illustrated in Figure 4.16, if the input sentence does not contain adequate information to extract triples, isolated concepts were extracted from them, assuming that these concepts incorporate useful information (Atapattu et al., 2012). In order to extract concepts, a set of regular expressions was defined using part-of-speech tags. The defined list of text patterns are shown in Table 4.11.

Table 4.11: Regular expression patterns to identify concepts; N: noun, J: adjective, b: beginning with, w: alphanumeric characters, s: space

Token size	Regular expression pattern	Description
4	<code>\bN\w+\s [4]</code>	Four consecutive nouns
	<code>\bJ\w+\s(\bN\w+\s)[3]</code>	Adjective followed by three consecutive nouns
	<code>(\bJ\w+\s)[2] (\bN\w+\s)[2]</code>	Two adjectives followed by two nouns
3	<code>\bN\w+\s [3]</code>	Three consecutive nouns
	<code>\bJ\w+\s(\bN\w+\s)[2]</code>	Adjective followed by two consecutive nouns
	<code>(\bJ\w+\s)[2]\bN\w+\s</code>	Two adjectives followed by a noun
2	<code>\bN\w+\s [2]</code>	Two consecutive nouns
	<code>(\bJ\w+\s)(\bN\w+\s)</code>	Adjective followed by a noun
1	<code>\bN\w+\s</code>	A noun
	<code>\bJ\w+\s</code>	An adjective

A greedy approach was adopted to extract concepts (see Appendix B).

A concept can occur in multiple levels within the lecture slide.

Local Synchronization Mechanisms

- * Locks
- * **Semaphores**
- * Barriers
- * Mutexes
- * ...

Semaphores

- * Simple abstraction to control access to a common resource
 - * signal and wait
- * Variable x that allows interaction via two operations
 - * `wait(x): while (x == 0) wait; --x;`
 - * `signal(x): ++x;`
- * Both operations are done atomically

Figure 4.22: Concepts occur in multiple levels of the slides

For instance, the concept *semaphores* occur both in title (level 1) and bullet point (level 2) (Figure 4.24). Therefore, it is required to determine the most suitable level for this concept.

Algorithm 4.3: Concept occurrence in multiple levels

```
Require: terms           // terms which have more than one occurrence
           document       // document that contains the 'term'
           root           // most general concept, usually the topic of the
document

1. Calculate the number of links each term contain
2. Select terms which contain maximum number of links
3. If number of maximum links greater than 1           // avoid disjoint
   concepts
   4a Calculate number of nodes from root to each term // proximity
   4b Select the term which has highest proximity
4. End if

Output: proximity       // most suitable level for the term
```

The usage of these isolated concepts will be discussed under 'hierarchy extraction'. This minimises the information loss even though these text contents do not produce triples.

Hierarchy

According to the definition of CMM by Villalon & Calvo (2008), in order to produce a topology, a hierarchical organisation of concepts is required, positioning the most general concept at the top and the most specific concepts arranged below (Novak & Canas, 2006). Based on the issues presented in Table 4.1, special text patterns (i.e. lexico syntactic patterns) are generally rare in any text, which makes it difficult to extract hierarchical relations using text (Cimiano, 2006). Instead, this thesis proposed following hierarchical relation patterns based on the natural layout of the presentation framework.

1. Slide title - bullet point - sub point (usually have a relation label)
2. Lecture topic - slide title - bullet point (usually have a relation label)
3. Lecture topic - slide title (mostly have no relation label)
4. Slide title - bullet point (mostly have no relation label)

An example slide with the corresponding hierarchical relationship (with and without relationship label) is shown in Figure 4.22 and 4.23 respectively.

The Process

- **Multiple parts**
 - Program code
 - Program counter
 - Stack
 - Data section
 - Heap
- **Program is passive entity, process is active**
 - Program becomes process when executable file loaded into memory

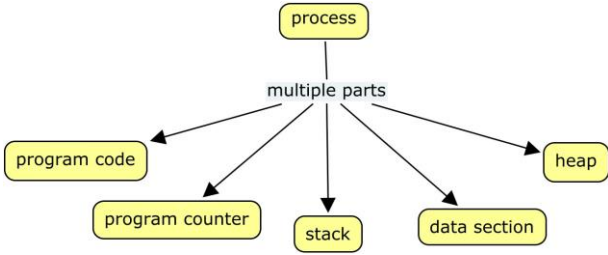


Figure 4.23: An example of hierarchy extraction (with ‘relation label’)

OSI Layers

- ⌘ Application
- ⌘ Presentation
- ⌘ Session
- ⌘ Transport
- ⌘ Network
- ⌘ Data Link
- ⌘ Physical

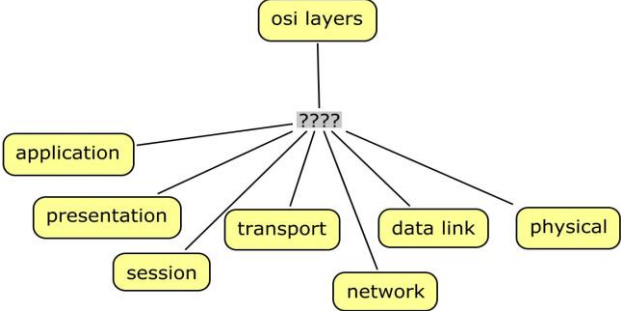


Figure 4.24: An example of hierarchy extraction (without ‘relationship label’)

Hierarchy extraction algorithm assigned a ‘level’ in order to simplify the discussion. Table 4.12 illustrates the levels which will be referred throughout the discussion.

Table 4.12: Hierarchy levels assigned for slide data

Level	Description	Examples using Figure 4.22 & 4.23
0	Lecture topic, generally title of the first slide	N/A
1	Slide title (each slide except first slide)	<i>Process, OSI Layers</i>
2	Bullet point (indentation level just after slide title)	<i>Multiple parts, application, presentation</i>
3 or greater	Sub points (enumerated bullet points)	<i>Program code, program counter</i>

In the example shown in Figure 4.22, the general concept (*root*) is considered from level 1 (e.g. *process*) and relation from level 2 (e.g. *multiple parts*); and specific concepts are those in level 3 (e.g. *program code, program counter*). This is similar to the hierarchical relation patterns defined in step 1 (or 2) above (i.e. *slide title - bullet point - sub point*). It is also possible to extract hierarchical relationships in a similar scenario between bullet points and enumerated sub points. However, it is not always common to have sub-points beyond level 3.

In the example shown in Figure 4.23, the general concept (*root*) is considered from level 1 (e.g. *OSI Layers*) and specific concepts are those from level 2 (e.g. *application*, *presentation*). However, these kinds of scenarios were not allowed to define relation labels (similar to steps 3 and 4). If the two participating concepts were ranked as important in the domain, it is unlikely that these concepts are removed from the generated concept maps even though they do not have a valid relation label. There is no existing approach to resolve this issue, unless introduce something similar to *Cloud* (Alves et al., 2001) which interacted with users to fix such issues. Instead, this research generated the hierarchy with the existing source of information by highlighting a blank label ‘????’ (see Figure 4.23). This allows domain experts to resolve the issue by providing a suitable label or simply removing the ‘????’ and restoring the relation without a label. Although, this lessens the level of automation, an intermediate participation of domain experts to assess the generated concept maps is suitable within the educational context, especially when these concept maps are utilised by learners. In addition, this process provided opportunity for domain experts to reflect on the conceptual overview of their lecture slides.

Algorithm 4.2: Hierarchy extraction with relation labels

```

Require: concept list           // in level 3 or higher

           concept list in parent // parent concepts list
           concept list in grandparent // grandparent concept list

1.If parent concept list and grandparent concept list not empty
2.For each concept in concept list
   For each grandparent_concept in grandparent concept list
     For each parent_concept in parent concept list
       Set start node of triple as grandparent_concept
       Set relation label of triple as parent_concept
       Set end node of triple as concept
     End for each
   End for each
3.End for each
4.End if

Output: triple           // triple with start and end node and relation label

```

The main motivation behind identifying these kinds of relations even without a label is to minimise the information loss. There are a larger number of isolated concepts in lecture slides which do not involve producing triples due to their incomplete nature, which is encouraged by the natural layout of the presentation framework.

Finally, hierarchy relations were checked against ‘cycles’ where start and end nodes in one triple repeat reversely in another triple.

4.2.5 Ranking

CMMF proposed three ranking models (i.e. *baseline*, *linguistic*, *structural*) based on hypotheses developed. Each of these ranking models were compared with human annotations to identify which ranking model best fits for ranking triples (Atapattu et al., 2014b).

Baseline Model

Lecture slides contain a natural layout of the presentation title (topic), slide titles, bullet points, and enumerated sub-points as already discussed in Section 4.2.4. This layout could be utilised to determine what information is important to the learners. For instance, ‘slide titles’ can be more important than text in ‘sub points’. In order to assess this assumption, baseline model integrated ‘text location’ in lecture slides (see Table 4.13).

Hypothesis 1: *Text location allocated by the natural layout of the presentation slides might influence human judgement of which concepts are most important*

Table 4.13: Concept importance determined by location

Location	Rank
Title or topic	3
Bullet statement	2
Sub point	1

Concepts which occur in multiple locations within the lecture slide were resolved using the algorithm 4.3.

Linguistic Feature Model

The hypothesis of the linguistic feature model was based on the recommendation of using the smallest number of words for a concept, usually a single word (Canas & Novak, 2009b). Table 4.14 shows the rankings based on the grammatical structure.

Hypothesis II: *Simple grammatical structures (nouns, noun phrases) of lecture slides might have higher influence than complex grammatical structures (nested sentences, dependent clauses, indirect objects) on human judgement of which concepts are most important*

Table 4.14: Concept importance determined by grammatical structure

Feature	Example grammatical structure	Rank
Noun phrase or verb phrase	(NP (NP (NNP Advantage)) (PP (IN of) (NP (NN unit) (NN testing))))	3
Simple sentence	(S (NP (NNP Process)) (VP (VBZ is) (NP (NP (NN program)) (PP (IN in) (NP (NN execution))))))	2
Complex sentence	(S (NP (DT A) (NN software) (NN process)) (VP (VBZ is) (NP (NP (DT a) (NN set)) (PP (IN of) (NP (NP (ADJP (RB partially) (VBN ordered)) (NNS activities)) (CC and) (NP (NP (JJ associated) (NNS results)) (SBAR (WHNP (WDT that)) (S (VP (VBP produce) (CC or) (VBP maintain) (NP (DT a) (NN software) (NN product))))))))))	1

As shown in the Table 4.14, complex sentences contain nested sentences (S), clauses (SBAR) and conjunctions (CC). Therefore, this thesis assumed these sentences contain definitions or elaborations rather than abstract concepts.

Structural Feature Model

The third candidate model integrated structural features (e.g. typographic information, term frequency and co-occurrence) that were observed throughout the presentation framework and graph-based features (e.g. degree centrality and proximity) which have been suggested by Leake et al. (2004) and Zouaq et al. (2012).

Hypothesis III: *Structural (degree centrality, proximity) and distributional (term frequency, degree of co-occurrence, typographic information) features might influence the human judgement of which concepts are most important.*

Log frequency weight

This measure calculated the number of occurrences of nouns and noun phrases. The term frequency (t_f) was normalised within a small range using ‘log frequency weight’. For instance, if the t_f is 1, weight will be 0.3 and for t_f of 100, weight will be 2.0. This prevented a bias towards

high frequency terms in determining important concepts. This will result in the term frequency being an important and influential factor in choosing important concepts rather than the only factor.

$$W_t = \log (1+ t_f)$$

Degree centrality (In-degree and out-degree)

The in-degree (n_i) and out-degree (n_o) of each node were calculated to determine the importance of that node within the concept map (Diestel, 2005). The ‘root’ node has higher out-degree, stressing its importance as the most general concept. Thus, those that have higher out-degree than in-degree are identified as of greater importance. Degree centrality is also substantial to identify disjoint concepts from central concept map. In the educational context, disjoint concepts do not facilitate meaningful learning. Therefore, degree centrality provides feedback for teachers to reflect on missing relations either on concept map generated or in the source document or both.

Degree of co-occurrence

According to Manning et al. (2008), if two key terms co-occur in many documents or pages in a document (equals to ‘slides’ in this corpus), it was assumed that those two terms have a strong relation, and hence, can be chosen as ‘domain concepts’. The degree of co-occurrence is measured using the *Jaccard coefficient*, a statistical measure which compares the similarity of two sample sets (Tan, Steinbach, & Kumar, 2005).

In order to measure the degree of co-occurrence between term t_1 and term t_2 , first calculate the number of slides, that t_1 and t_2 co-occurs. This is denoted as $|n_1 \cap n_2|$. Then calculate the number of slides the term t_1 ($|n_1|$), t_2 ($|n_2|$) occurs. The degree of co-occurrence of t_1 and t_2 is denoted by $J(t_1, t_2)$ and is given by,

$$J(t_1, t_2) = \frac{|n_1 \cap n_2|}{|n_1 \cup n_2|} = \frac{|n_1 \cap n_2|}{(|n_1| + |n_2| - |n_1 \cap n_2|)}$$

This value was utilised as a key decisive factor for noise detection in slide-levels since terms such as *course announcements* or *references* have low degree of co-occurrence with other important domain concepts.

Typographic information

Lecture slides often contain emphasised text (e.g. different font colour, underlined) to illustrate the importance of particular concepts in the given domain. This thesis introduced a probability model to select candidate concepts with the use of their typographic information. According to the proposed model, terms which contained *infrequent styles* were allocated higher weights, and hence, considered as more important. For instance, the font colour of lecture slides were set as ‘black’ while the font colour of few concepts were changed to ‘red’ to emphasise their importance. An example to calculate weights associated with typographical information is included in Appendix B.

Proximity

CMMF considered the ‘lecture topic’ as the *root* (or central concept) of the concept map. Therefore, it was assumed that the concepts that have a higher proximity to the *root* are expected to be more important than those with lower proximity (Leake et al., 2004). The proximity weight (W_p) is obtained by calculating the number of nodes (d_n) from *root* to the participating node (inclusive).

$$W_p = \frac{1}{d_n}$$

4.2.6 Concept Map Visualisation

In order to complete the CMM process, the extracted triples need to be visualised as a concept map. CMapTools developed by the Institute for Human and Machine Cognition (IHMC) is capable of importing triples written using ‘propositions as text’, ‘CMap outline’, ‘CXL’, ‘XTM/XCM’ or ‘IVML’ format (Canas, Hill, et al., 2004). CMMF converted the triples into CXL (Concept Map Extensible Language) format (Canas, Hill, et al., 2004), an XML-based, light-weight file format to store concept maps. Table 4.15 illustrates the most commonly used elements and attributes of CXL.

Table 4.15: Elements and attributes of CXL

CXL element	Attributes	Parent	Description
cmap	none	none	Main element
map	root-id width height	cmap	Defines the structure of the map
concept-list	none	map	List of concepts in the map
linking-phrase-list	none	map	List of linking phrases in the map
connection-list	none	map	List of connections in the map
concept	id label	concept-list	Defines the concept
linking-phrase	id label	linking-phrase-list	Defines the linking phrase
connection	id from-id to-id	connection-list	Defines the connection

Figure 4.25 shows the structure of an example CXL file.

```
<?xml version="1.0" encoding="UTF-8"?>
<cmap xmlns:dcterms="http://purl.org/dc/terms/" xmlns="http://cmap.ihmc.us/xml/cmap/"
  xmlns:dc="http://purl.org/dc/elements/1.1/" xmlns:vcard="http://www.w3.org/2001/vcard-rdf/3.0#">
  <map width="426" height="179">
    <concept-list>
      <concept id="1NODPKDCB-15CZP60-BD" label="data section"/>
      <concept id="1NODPJ7JT-R70BKR-6W" label="program code"/>
      <concept id="1NODPHYMC-1CVLWBV-62" label="process"/>
      <concept id="1NODPJSBZ-2CHS7GW-8P" label="program counter"/>
      <concept id="1NODPK436-1K1T1T5-9P" label="stack"/>
      <concept id="1NODPKM14-1TMPPC3-C9" label="heap"/>
    </concept-list>
    <linking-phrase-list>
      <linking-phrase id="1NODPJ7KB-1G5F69Y-6Z" label="multiple parts"/>
    </linking-phrase-list>
    <connection-list>
      <connection id="1NODPKM1D-2DRTP65-CD" from-id="1NODPJ7KB-1G5F69Y-6Z" to-id="1NODPKM14-1TMPPC3-C9"/>
      <connection id="1NODPKDCL-1GH4S1X-BH" from-id="1NODPJ7KB-1G5F69Y-6Z" to-id="1NODPKDCB-15CZP60-BD"/>
      <connection id="1NODPJ7KM-1CRGYKD-79" from-id="1NODPJ7KB-1G5F69Y-6Z" to-id="1NODPJ7JT-R70BKR-6W"/>
      <connection id="1NODPJSC9-16H9Y42-8S" from-id="1NODPJ7KB-1G5F69Y-6Z" to-id="1NODPJSBZ-2CHS7GW-8P"/>
      <connection id="1NODPK43S-4NLVCV-9S" from-id="1NODPJ7KB-1G5F69Y-6Z" to-id="1NODPK436-1K1T1T5-9P"/>
      <connection id="1NODPJ7KL-WVWF89-73" from-id="1NODPHYMC-1CVLWBV-62" to-id="1NODPJ7KB-1G5F69Y-6Z"/>
    </connection-list>
  </map>
</cmap>
```

Figure 4.25: Sample CXL file

The generated CXL file can be directly imported to CMapTools using the option 'File -> import -> CMap from CXL file'. Figure 4.26 shows an example CourseCMap generated from the *Software Engineering* course using CMMF. CMapTools provides a simple, user friendly interface for concept mapping, including auto layout, editing, sharing in the web, attaching related resources and merging with other concept maps. The concept maps constructed or generated from CMapTools also allow exporting the map as an image file.

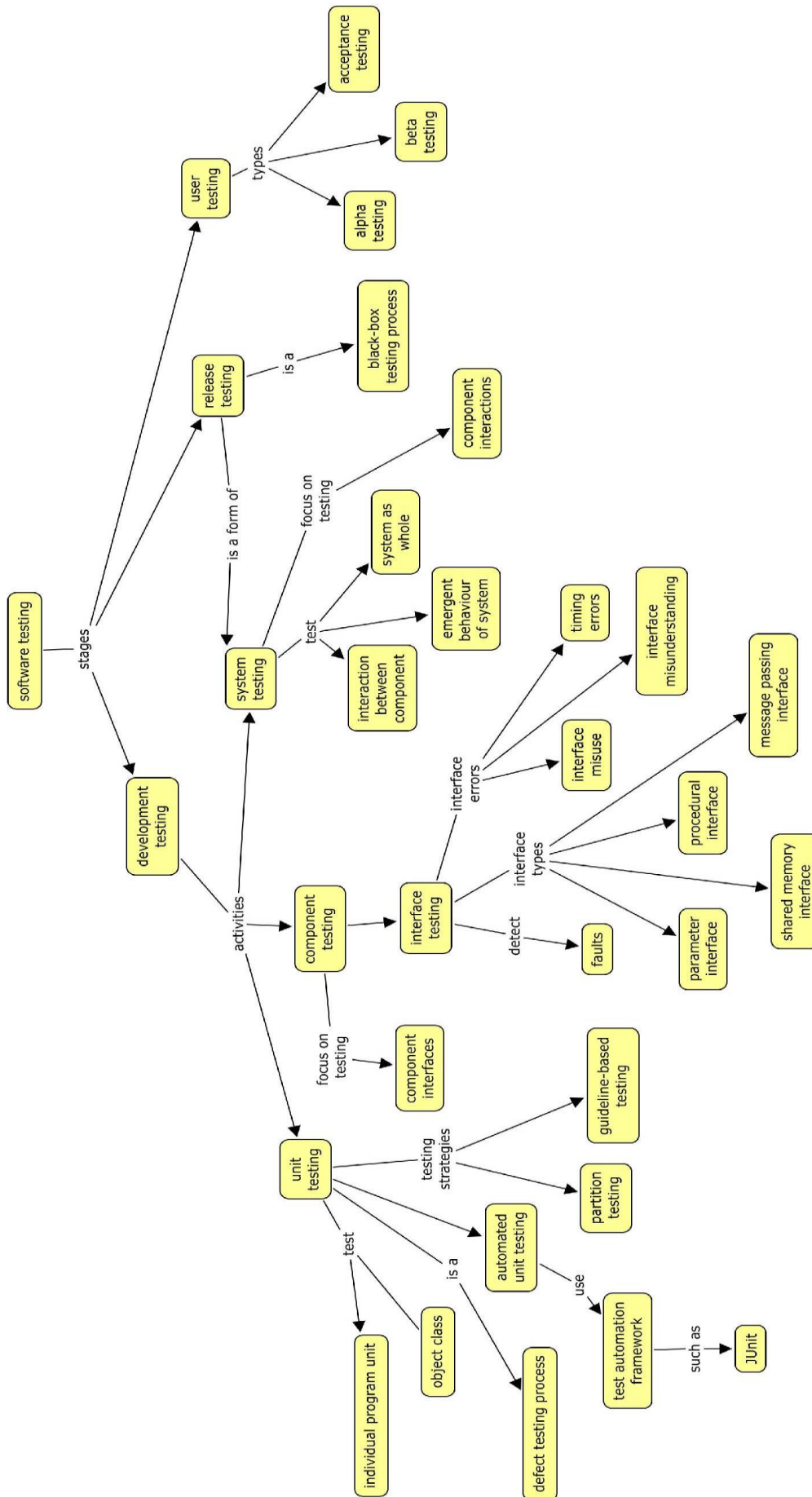


Figure 4.26: An example CourseMap

4.2.7 Summary

This chapter presented the design and development of the concept map mining framework (CMMF). The main objective of this thesis is to extract high quality concept maps from lecture slides, enabling them to be utilised as scaffolding in the problem solving context.

Based on the CMM definition by Villalon & Calvo (2008), this thesis categorised the concept map mining process into four stages: structured data analysis, knowledge acquisition, ranking and visualisation. Initially, slides which contained information inappropriate (e.g. *course announcements, references*) for inclusion in a concept maps were identified and removed. CMMF utilised contextual features specific to the presentation framework to resolve syntactically and semantically missing and ambiguous sentences. In the second stage, the useful knowledge was extracted in the form of concept-relation-concept triples using NLP-based algorithms. The extracted knowledge was arranged in a hierarchy based on the natural layout of the presentation framework. The ranking stage utilised three models to rank the triples according to their importance. Baseline model utilised the ‘location’ of text within a lecture slide to determine the importance. Linguistic feature model considered the ‘grammatical structure’ of text to determine the importance of concepts. Structural feature model utilised structural and graph-based features (e.g. *term frequency, degree centrality, proximity*) to rank the triples according to their importance. The final stage of CMMF visualised extracted triples using IHMC CMapTools with the use of concept map extensible language (CXL) (Canas, Hill, et al., 2004).

The development of CMMF contributed to several technical advancements in the field of research. Due to the issues associated with lecture slides such as element of noise, the application of previously developed algorithms for knowledge acquisition from lecture slides was undesirable. In order to fill this gap, CMMF introduced approaches for automated noise detection and elimination, pronoun and determiner resolution, and transforming incomplete sentence fragments into complete sentences using the contextual features embedded in the presentation framework. More importantly CMMF was capable of extracting relation labels between concepts which fulfilled the motivation of facilitating meaningful learning.

Apart from the contributions to knowledge acquisition, the CMMF developed an evaluation methodology to overcome the issues associated with binary classification system which evaluates the extracted information as *relevant* or *non-relevant*. The new methodology considered the *relevance* measure of as a rank, with some highly important knowledge, averagely important knowledge and others with low importance.

The next chapter (Chapter 5) presents the usage of concept maps generated using CMMF as scaffolding resources to facilitate problem solving.

Chapter 5

Task-adapted Knowledge Organisation

This thesis initially discusses the design and development of concept map mining framework (CMMF). This chapter discusses the adoption of concept maps extracted using CMMF to provide additional scaffolding in the problem solving context, in the case where students are lacking the required skills.

Even though problem solving with the use of concept maps as scaffolding has been studied previously, those studies did not specifically focus on information relevant to each problem. To fill this gap, this thesis investigated an approach to provide the most relevant information for problem solving, particularly in answering question in an online environments using concept maps generate from lecture slides. The concept maps utilised as scaffolding in this research were extracted based on each question. This process is known as ‘*task-adapted knowledge organisation*’. Questions in this context are similar to formative or summative questions provided to engage and motivate students, focus attention, guide learning, provide opportunity for practice and self-assessment (Dillon, 1988; Hunkins, 1972; Wilen, 1986). The idea of task-adapted knowledge organisation is illustrated using examples in Figures 5.1 and 5.2.

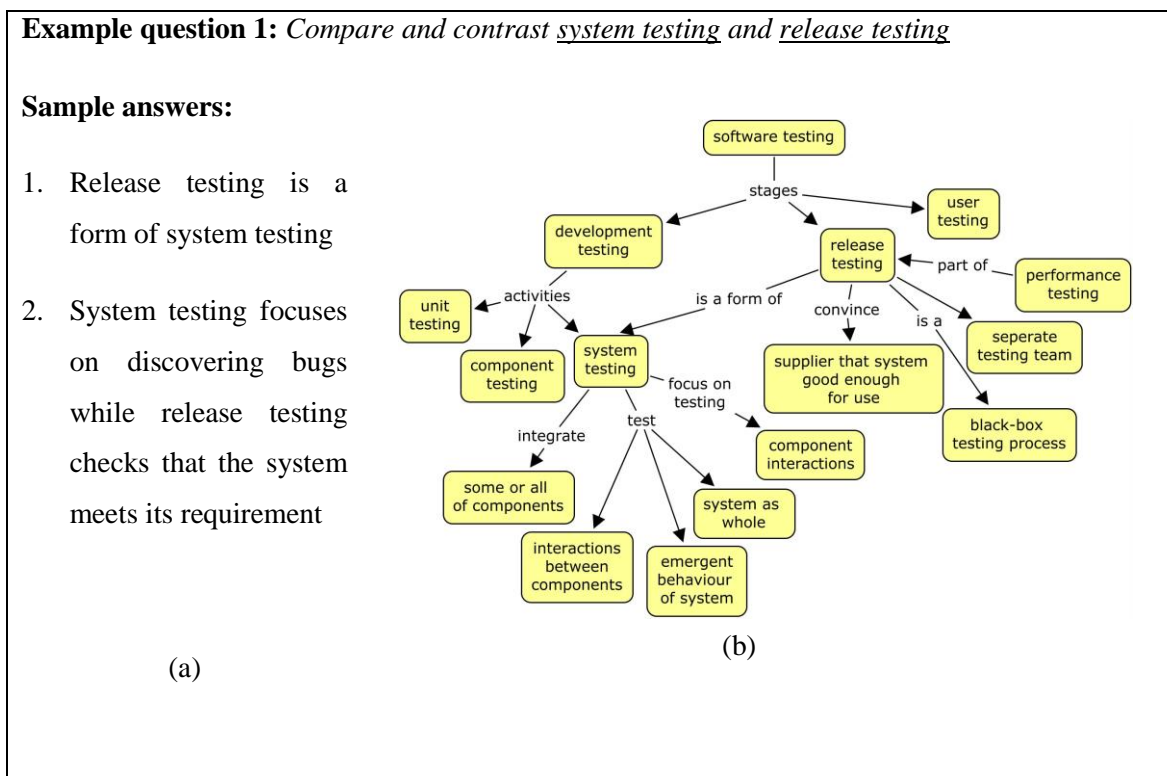
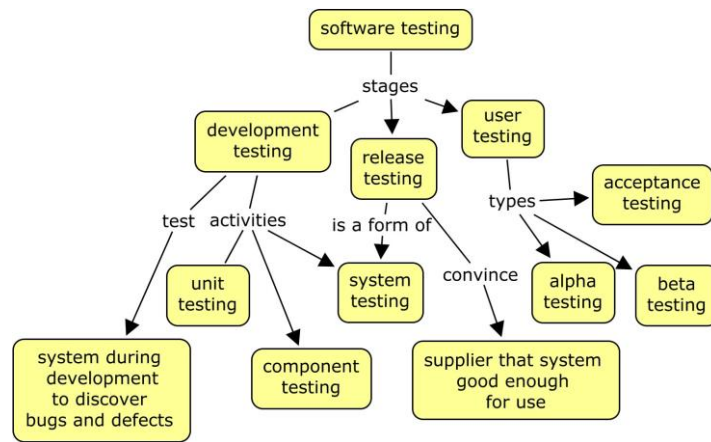


Figure 5.1: Example question 1 (a) text-based answer (b) task-adapted concept map

Example question 2: *Development testing is a software testing stage which includes all testing activities carried out by the development team. Identify other stages of Software testing.*

Sample answers:

1. Release testing
2. User testing



(a)

(b)

Figure 5.2: Example question 2 (a) text-based answer (b) task-adapted concept map

When students lack the required skills to answer questions in an online environment, the typical way of obtaining additional scaffolding is by referring to educational materials such as text books, lecture slides or lecture videos. The information presented in these materials is linear by nature, which does not aid the learners in identifying relationships between new and already known information, resulting in poor support for knowledge organisation (Brandt et al., 2001; Kinchin, 2006a). To clarify this further using the example question in Figure 5.1: in order to compare two concepts, the student should know both the concepts, how each of these concepts are related, and are also different from each other. Unless they are clearly defined in the material or demonstrated in the lecture, it is difficult to identify such relationships in sequentially-structured educational materials as already discussed in Section 2.4 (Kinchin et al., 2008; P. A. Martin, 2008). Therefore, problem solving using task-adapted concept maps as scaffolding supports effective knowledge organisation to improve learning outcome.

The effect of task-adapted concept maps as scaffolding for answering questions was supported by the empirical studies of Canas & Novak (2006), Dias & Sousa (1997), Edmunds & Morris (2000) and Eylon & Reif (1984). These studies analysed texts according to their underlying hierarchies, and showed that information contained in higher levels of the hierarchy was recalled better than information in the lower levels (Kintsch & Keenan, 1974; Meyer, Brandt, & Bluth, 1980). Based on these findings, a study by Eylon & Reif (1984) demonstrated that the students who received task-adapted information performed significantly better than the control

group who received hierarchical information without task adaptation in the performance tasks. Even though this previous study provided a basis for this research, it is not a feasible approach to organise the knowledge according to tasks *manually* (Eylon & Reif, 1984). Thus, this thesis takes a new perspective to develop a machine-based framework for *task adaptation*.

According to Canas & Novak (2006), a concept map can be developed in two ways: either as a general map to represent knowledge in a domain/topic or as an answer to a ‘*focus*’ question. The former approach is similar to that used in concept maps generated from lecture slides using CMMF which represents the domain. However, the latter approach is more effective since building a concept map to answer a question involves more dynamic thinking and a deeper understanding (Canas & Novak, 2006). For instance, constructing a concept map to answer ‘*what are birds?*’ is more important than ‘*create a concept map about birds*’ (Canas & Novak, 2006).

Providing more relevant information as scaffolding will motivate students by reducing the information overload which can lessen the anxiety, stress, alienation and learning disorientation among learners (Dias & Sousa, 1997; Edmunds & Morris, 2000).

To measure the effectiveness of task-adapted concept maps, the hypothesis was constructed as ‘*students who receive task-adapted concept maps as scaffolding will have an increased learning gain compared to those who did not receive task-adapted concept maps*’.

To the best of the authors’ knowledge, this is the first study which facilitates problem solving with scaffolding via task-adapted concept maps in an *automated manner*. Previously, research had been carried out to return semantic networks to answer general questions within the information retrieval context (Dali, Rusu, Fortuna, Mladenec, & Grobelnik, 2009). However, their work was limited to return the semantic graph of the *whole document* which contains the answer, in contrast to the *task-adapted semantic graphs*.

5.1 Design of the Task-Adapted Scaffolding Framework

When introducing a new software framework within the educational context, it should be designed according to underlying learning theories (Tchounikine, 2011). This research of introducing concept maps for problem solving is grounded in the cognitive learning theories (Ausubel, 1968; Novak & Gowin, 1984). The scaffolding as a learning technique to support students to achieve their learning goals is grounded in the Social Constructivism Theory of Vygotsky (1978) and his concept known as the Zone of Proximal Development (Beed et al., 1991; Dabbagh, 2003).

Teachers' acceptance of new design is a critical issue. The new design should be flexible enough for teachers to use without additional workload. This has been achieved through utilising the quizzes that have already been constructed for self-assessments within a Learning Management System (LMS). A Learning Management System is a software application used for the delivery of e-learning education courses (e.g. Moodle, Blackboard). The existing quizzes allow the teacher to design and build a large variety of question types including multiple choice, short answer, essay, calculations and close tests. The teacher can select how questions behave during the quiz. It can be like an assessment, where a student gets no feedback while attempting. Alternatively, the feedback can be revealed to the student, and provides them multiple attempts to answer the question after having read the feedback (Moodle, 2014).

Unlike other works by Dali et al. (2009) and Lopez et al. (2007) which supported only a restricted grammar, this research supported both interrogative (i.e. starts with *wh*-clause) and imperative (i.e. starts with *identify*, *define*) questions without restrictions in the grammatical structure. Further, the framework was capable of processing not only the 'question text', but also any supporting 'question descriptions'. Some questions contain a 'question description' prior to presenting the actual question text. These are usually a single or multiple sentences describing the context of the question. Thus, teachers are not expected to construct new quizzes that are compatible with the new framework.

Another motivation behind designing this framework was to improve students' learning experience without presenting an additional burden. In order to fulfil this motivation, the framework was designed as an enhancement to an existing 'quizzes' of the LMS. These quizzes provide opportunities for practice and self-assessment and receive feedback about performance which considers as an important part of a learning environment.

Architecture of the Task-adapted Scaffolding Framework

CourseCMap as extracted using CMMF (Chapter 4) can be utilised as the domain model, supporting the extraction of task-adapted concept maps based on a set of input questions. The term ‘*TASF*’ will be used to describe the framework for task-adapted scaffolding. The term ‘*TSKCMMap*’ will be used to describe task-adapted concept maps extracting using *TASF*.

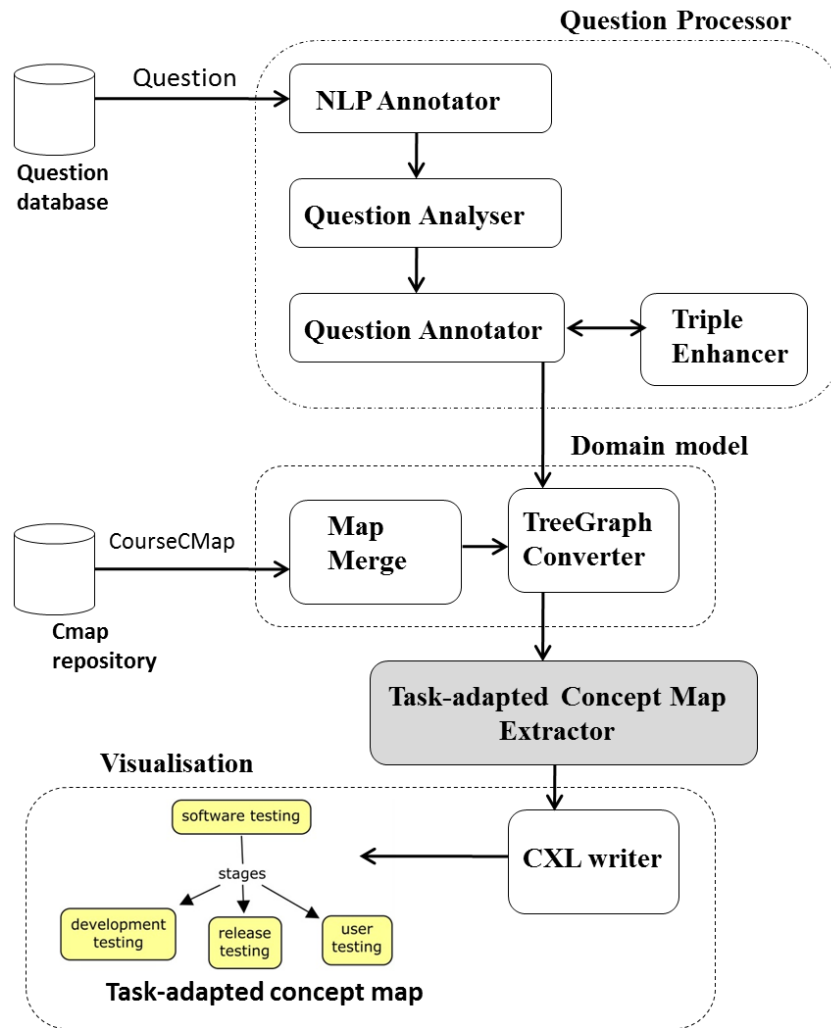


Figure 5.3: High-level architecture of the task-adapted scaffolding framework

As illustrated in the architecture diagram (Figure 5.3), *TASF* is categorised into four components: question processor, domain model, task-adapted concept map extractor and visualisation.

Prior to explaining the design considerations of each of these components, a comparison between *TASF* and ‘question answering’ system is presented since both of them share similar logic to process input questions but produce a different output. Question answering (QA) is the

process of automatically answering the questions submitted by humans in natural language. QA is a problem in the field of information retrieval and natural language processing (Jurafsky & Martin, 2009).

QA systems process the input question about facts (a factoid question) (e.g. *What is the capital of Australia?*) and return a text answer as output while the intention of *TASF* was to return task-adapted concept maps to improve learning outcome in the educational context. Table 5.1 presents a comparison of *TASF* and QA system.

Table 5.1: Comparison between the components of typical QA system and *TASF*

Component	QA system	Task-adapted scaffolding framework
Question processor	Identify the ‘question type’ and obtain set of key words (known as ‘query’) included in the question to extract answers	Identify the ‘question type’ and obtain triples from the question to extract task-adapted concept maps
Domain model	Collection of documents	Collection of concept maps extracted using CMMF
Answer extractor	Extract text answers that best matched with the query	Extract task-adapted concept maps that overlaps with triples of the questions
Visualisation	None	As a concept map

Question processor

The main aim of the ‘question processor’ was to automatically identify the ‘question type’ and convert the input question into ‘triple’ form.

Question types are different based on the context they are utilising. In QA systems, question types should be identified automatically in order to match with the expected answer type (see examples in Table 5.2).

Table 5.2: Question types defined in QA systems (Dali, et al., 2009; Kolomiyets & Moens, 2011; Olney, et al., 2012)

Question type	Question stem	Expected answer type
Factoid	Who	Person
Factoid	When	Date/time
Factoid	Where	Location
List	What	List
Quantification	How many	Number
Verification	Do, Is	Yes/No

As discussed in the Section 2.5 (see Table 2.4 and 5.2), the majority of the question types defined in QA systems are restricted to factoid questions where the answer is a fact, list or a

definition. Even though QA systems and TASF have the same underlying mechanism for question type identification, the restriction of QA systems into fact-based questions is not suitable for the educational context, since they only measure the ‘recall’ skills of learners (Dali et al., 2009).

In the educational domain, question type is generally defined according an educational taxonomy such as the cognitive domain of Bloom’s taxonomy (Bloom, Englehart, Furst, Hill, & Krathwohl, 1956). The construction of questions according to the learning objectives of Bloom’s taxonomy utilises keywords presented in Table 2.3.

By considering the limitations of previous works related to QA systems and the requirement of adopting education taxonomies, this research carried out a background study to investigate the question types suitable for TASF. This study analysed Software Engineering examination questions from year 2000 to 2012. The examinations consisted of 60 broad questions, the majority of which consisted of multiple sub-questions, which assessed different learning objectives. In such situations, the analysis considered them as separate questions. This study considered only the questions based on lecture material, with other types of questions such as scenario-based eliminated. The selected set of questions (approximately 100) were categorised based on the levels of Bloom’s taxonomy (Bloom et al., 1956). Concept maps were drawn manually using IHMC CMap tools (Canas, Hill, et al., 2004) by considering the examination questions and the answers obtained from the corresponding marking schemes. A computer-based algorithm compared the overlap between manually drawn concept maps of questions and answers (known as ‘answer map’) with CourseCMap of corresponding Software Engineering topic. The results of this stage of the study confirmed that the idea of ‘task-adapted concept maps’ is possible.

Further, based on the number of concept in the answer map, distance between concepts of questions and answers and the decision to include the siblings, parents or children of overlapping concepts, this research defined two question types (‘descriptive’ and ‘comparison’) to be utilised in the TASF (see Table 5.3). Although, there are six objective levels in Bloom’s taxonomy which could lead to six different question types, some of those levels share the same keywords (e.g. *modify* in ‘application’ and ‘synthesis’ level, *explain* in ‘synthesis’ and ‘comprehension’ levels - see Table 2.3). Since TASF did not place restrictions on the grammatical structure of the question, it was challenging to automatically identify six question types when some of the question types share the same keywords.

Table 5.3: Question types supported by TASF

Level	Question type	Keywords
Knowledge	Descriptive	Identify, define, list, describe
Comprehension	Descriptive	Distinguish, explain, summarise
Application	Descriptive	Apply, demonstrate, discover
Analysis	Comparison	Compare, contrast, analyse
Synthesis	Descriptive	Categorise, create, organise
Evaluation	Descriptive	Justify, evaluate, criticise

According to the question types defined in the TASF (Table 5.3), ‘comparison’ type question included keywords ‘*compare, contrast or analyse*’. This level was expected to return a higher number of concepts in the task-adapted concept maps in order to provide adequate context to compare multiple concepts in contrast to other ‘descriptive’ levels. The remaining questions were categorised as ‘descriptive’.

As in previous work of Dali et al. (2009), the key words of the ‘comparison’ type were not expected to be at the beginning of the question (i.e. question stem). They could be included anywhere in the question text without affecting the creativity or flexibility in question construction. Additionally, TASF was capable of processing both interrogative type questions (e.g. *what, why*) and imperative constructs (e.g. *identify, define*) as illustrated in examples below.

Example question 1 [type = comparison]: *What are the advantages and disadvantages of waterfall model compared to other process models?*

Example question 2 [type = descriptive]: *Identify the types of interfaces in component testing.*

Example question 3 [type = comparison]: *Compare and contrast White-box testing and Black-box testing.*

Example question 4 [type = descriptive]: *What is mean by Test-Driven Development?*

The conversion of the question into ‘triple’ form was performed using a sub component called ‘question annotator’. The general idea behind converting the question into triple form (known as ‘question triples’) was matching the similarity with the triples included in the domain model (known as ‘domain triples’) more efficiently. In QA systems, this is similar to indexing the questions using a bag-of-words model or similar representation and applying the same technique to index the document collection for efficient retrieval of text answers. Previous work

in utilising domain triples to answer natural language questions was discussed in the START system (Katz, 1997) and AnswerArt (Dali et al., 2009; Dali, Rusu, Fortuna, Mladenić, & Grobelnik, 2010).

Unlike in the CMMF which generated all the three elements of triple to produce concept maps, it was not expected to extract all the elements of question triples (see examples below). Question triples should contain one or more elements including at least one concept.

Sample question 1 [type = descriptive]: *What is caching?*

Question triple: *(caching, is, ?)*

Sample question 2 [type = comparison]: *Compare and contrast white-box testing and black-box testing*

Question triple: *(white-box testing, black-box testing, ?)*

Domain Model

Task-adapted concept map extraction utilised the CourseCMap generated using CMMF as the domain model (see Figure 4.26). Generally, a single concept map was generated from a lecture topic in order to reduce the complexity of the map. However, when these maps were utilised to model the domain knowledge, concept maps with related information were merged in order to create a rich knowledge source by creating cross-links between different parts of the overall map.

Task-adapted Concept Map Extractor

Task-adapted concept map extraction was the core process of TASF. The purpose of this component was to extract task-adapted concept maps based on the ‘question type’ and the ‘question triples’.

Task-adapted concept map extraction from a CourseCMap can be identified as a problem of ‘sub graph matching’ which extracts ‘sub-graphs’ from a ‘graph’ according to an input pattern (Gallagher, 2006). According to Gallagher (2006), the ‘sub-graph matching’ problem can be defined as follows;

A data graph $G = (V, E)$, composed of a set of vertices V and a set of edges E . Each $e \in E$ is a pair (v_i, v_j) where $v_i, v_j \in V$. The vertices and/or edges of G may be typed and/or attributed

A pattern graph (or ‘question triple’ in TASF) $P = (Vp, Ep)$, which specifies the structural and semantic requirements that a sub graph of G must satisfy in order to match the pattern P .

The task is to find the set M of sub graphs of G that ‘match’ the pattern P . A graph $G' = (V', E')$ is a sub graph of G if and only if $V' \subseteq V$ and $E' \subseteq E$.

The most common approach of sub graph matching is the ‘structural matching’ approach which is known as ‘sub graph isomorphism’ (Ullmann, 1976). This approach generally operated on the structure of two graphs such as number of edges adjacent to vertices, but did not consider the *labels* and attributes of vertices. In this thesis context, vertices and edges are similar to concepts and relations respectively. In TASF, labels of concepts and relations were key factors to determine the task-adapted concept maps based on the question triples. Thus, structure-based sub graph matching was not suitable to solve the problem in this thesis.

‘Similarity-based sub graph extraction’ approach is therefore considered. Previous works in the area of similarity-based sub graph extraction involved inexact matching where the matching algorithm returned a ranked list of most similar matchings, but not the exact matchings (Gallagher, 2006). Generally, wildcard patterns were utilised to perform these kinds of matchings. This led to returning imprecise sub graphs in the educational context. For instance, a question ‘*What is a generic software process model?*’ produced the triple (*generic software process model, is, ?*). In this question, the inexact similarity matching might consider (*generic software process model, is, ?*), (*software process model, is, ?*), (*process model, is, ?*), (*process, is, ?*), (*model, is, ?*) and (*software process, is, ?*) using wildcard patterns. Each of these question triples produced using wildcard patterns were also probable domain triples of the topic ‘Software Process’. Therefore, if there was no exact matching with (*generic software process model, is*), the returning of other sub-graphs was imprecise within the problem of ‘task-adapted concept map extraction’. Alternatively, when there was no exact match of question triples with *CourseCMap*, TASF incorporated triples from ‘question description’ to extract sub-graphs. Therefore, in TASF, there will be only *exact matches* between question triples and domain triples of *CourseCMap*.

TASF supported ‘exact similarity matching’ using two techniques. First, the labels of concepts and relations of questions were mapped to their base form using the lemmatisation technique. Similar techniques were applied to CMMF as discussed in Section 4.2.2. Secondly, an external lexical database called *WordNet* was utilised (Miller et al., 1990). This ensured the enhancement

of question triples through the use of synonyms. Therefore, TASF developed techniques to extract task-adapted concept maps using ‘similarity-based exact sub graph matching’ technique with the use of some of the related works of Dali et al. (2009).

The Figure 5.4 illustrates the task-adapted concept map extraction process using a subset of a CourseCMap.

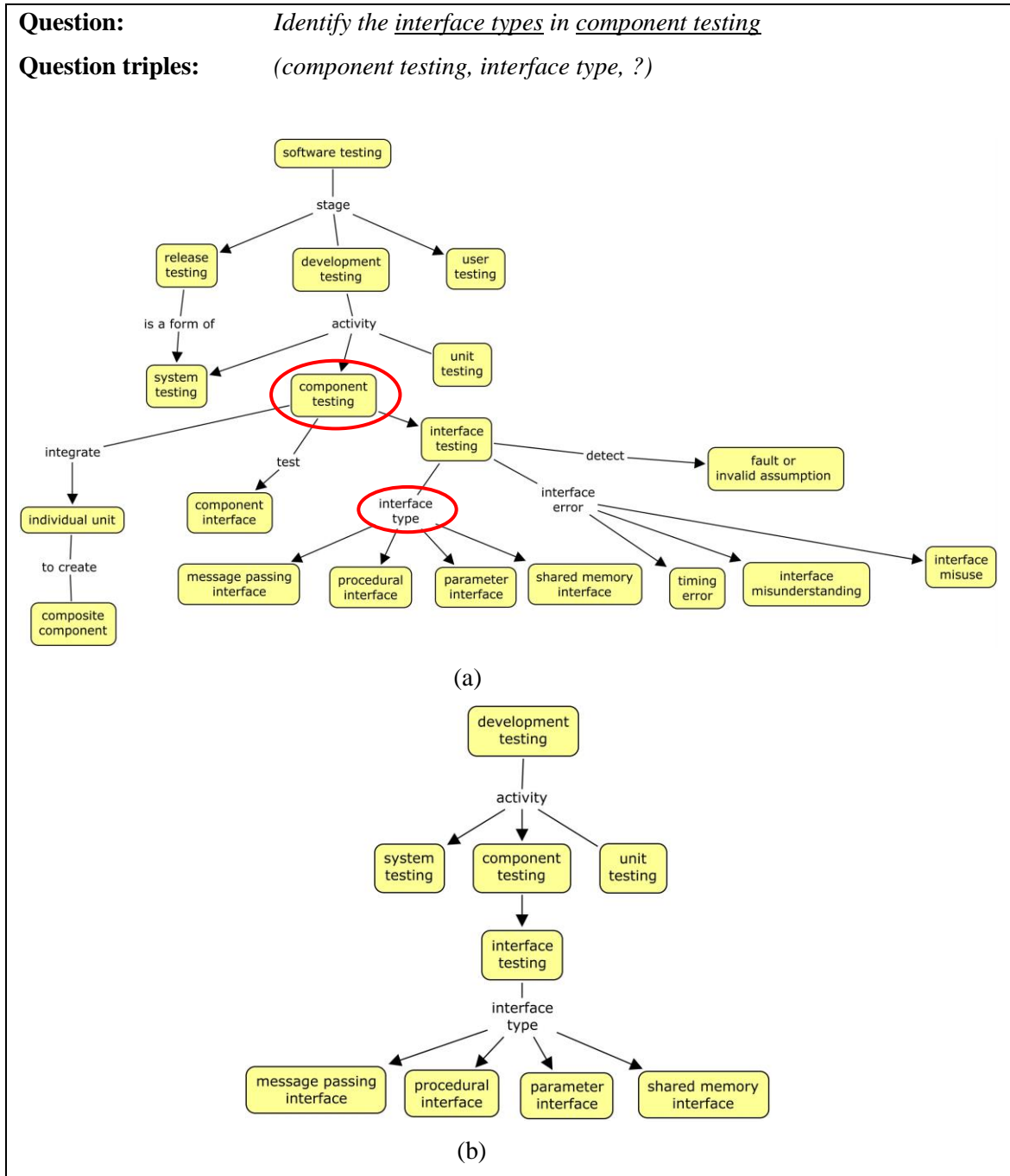


Figure 5.4: (a) CourseCMap (b) TSKCMap

5.2 Development of the Task-Adapted Scaffolding Framework

5.2.1 Development Environment

The framework was written using the Java language (version 1.7). The extracted TSKCMaps were stored as XML-based concept map extensible language (CXL). Similar to CMMF, this work utilised NLP tools and libraries to process questions such as Stanford Core NLP tools (Klein & Manning, 2003a, 2003b; Manning et al., 2014).

WordNet, a lexical database of English (version 2.1 for Windows) (Miller et al., 1990) was integrated to identify synonyms of question triples. For instance, a question ‘*What are the stages of Software testing?*’ produces triples (*Software testing*, *stages*, ?). CourseCMap might contains triples such as (*Software testing*, *phases*, *Development testing*), (*Software testing*, *phases*, *Release testing*). The SynSet of WordNet database identifies both verbs ‘*phases*’ and ‘*stages*’ are as synonyms, which improved the similarity-based exact sub-graph matching between question triples and domain triples.

5.2.2 Natural Language Annotation

NLP annotation included parsing the input question using NLP tools (Klein & Manning, 2003a, 2003b; Manning et al., 2014). The use of the grammatical structure of the question was more accurate in contrast to utilising models like bag-of-words (Salton & McGill, 1986). The algorithm for NLP annotation has already been discussed in Section 4.2.2. However, Penn Treebank tags for *questions* are different from *sentence* texts (Marcus et al., 1994). Therefore, Table 5.4 and Figure 5.5 illustrate an example question ‘*What is the purpose of Regression testing?*’ using NLP annotations and parser tree respectively.

Table 5.4: NLP annotations of an example question

Question	What	is	the	purpose	of	regression	testing
Part-of-speech	WP	VBZ	DT	NN	IN	NN	NN
Lemma	What	be	the	purpose	of	regression	testing

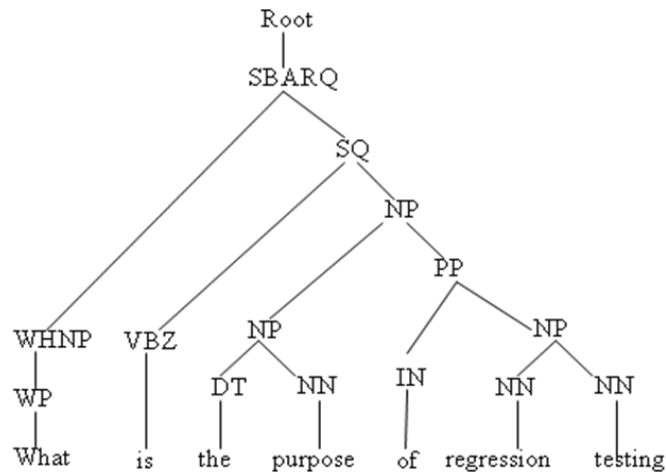


Figure 5.5: Parser tree of an example question

Further, Table 5.5 illustrates the Penn Treebank tags used to process *questions* (Marcus et al., 1994).

Table 5.5: Penn Treebank tags for question texts

Level	Tag	Description
Clause	S	Simple declarative clause
	SBAR	Clause introduced by subordinating conjunction
	SBARQ	Direct question introduced by <i>wh</i> -word or a <i>wh</i> -phrase
	SINV	Inverted declarative question
	SQ	Inverted yes/no question or main clause of <i>wh</i> -question
Phrase	WHADJP	<i>Wh</i> -adjective phrase
	WHAVP	<i>Wh</i> -adverb phrase
	WHNP	<i>Wh</i> -noun phrase
	WHPP	<i>Wh</i> -prepositional phrase
Word	WDT	<i>Wh</i> -determiner
	WP	<i>Wh</i> -pronoun
	WP\$	<i>Wh</i> -possessive <i>wh</i> -pronoun

5.2.3 Question Processor

This component converted the input ‘question’ into ‘triple’ form, allowing comparison between question triples and domain triples. This involved four steps; obtaining the grammatical tree of the question using linguistic parsers, identifying the ‘question type’ using the question analyser, extracting triples using the question annotator and finally, enhancing triples using the triple enhancer.

Question Analyser

The aim of the question analyser was to automatically identify the ‘question type’. The definition of question type (see Table 5.3) with respect to the educational context has already been discussed in Section 5.1. The automated identification of question type was straightforward since TASF had only two question types. The questions which contained the

key words ‘*compare*’, ‘*contrast*’, ‘*analysis*’ were identified as ‘comparison’ while the remaining questions were categorised as ‘descriptive’.

Question Annotator

The aim of the question annotator was to transform the input question into ‘triple’ form. This component had two stages. The first stage identified the triples from ‘question text’. These triples acted as main inputs for TSKCMap extraction. The second stage extracted all possible combinations of triples from ‘question description’. Afterward, if the question text had not included adequate information for TSKCMap extraction, the probable triples from the question description were utilised.

The latter process of identifying triples (either as subject-verb-object or noun-verb-noun) from question description is similar to the triple extraction from English sentences which was already explained in Section 4.2.4. The process of triple extracting from question text starting with interrogative words (e.g. *wh*-clause) was different. This reused some of the heuristics proposed by Dali et al. (2009) and incorporated specific question types to propose the algorithm 5.1. Figure 5.6 illustrates a skeleton question tree which used to describe the algorithm.

Algorithm 5.1: Triple identification from Questions

```

Require: Question tree           // Parse tree of the question
           Question type           // either 'comparison' or 'descriptive'

1.If question type is 'comparison'
  1. Obtain smallest VP sub tree
  2. Extract leaves of first pre terminal of NP sub tree in VP sub
    tree as C1
  3. Extract leaves of second pre terminal of NP sub tree in VP sub
    tree as C2
  4. Triple is set as (C1, C2, ?)
2.Else if TN equals to SQ           // refer to Figure 5.6
  1. if N0 equals to VB and N1 equals to NP
    i. if N3 equals to NP and N4 equals to PP
      Triple is set as (obj (N4), N3, ?)
    ii. end if
  2. end if
3.Else if TN equals to SBARQ then
  1. If N0 equals to WHNP and N1 equals to SQ
    i. If N3 equals to VP
      Triple is set as (?, N3, obj (N3))
    ii. Else if N4 equals to NP and N5 equals to VP
      Triple is set as (N4, N5, ?)
    iii. End if
  2. End if
  3. If N0 equals to WHADVP and N1 equals to SQ and N4 equals to NP
    Triple is set as (N4, ?, ?)
  4. End if
4.End if

Output: triple           // triple extracted from question

```

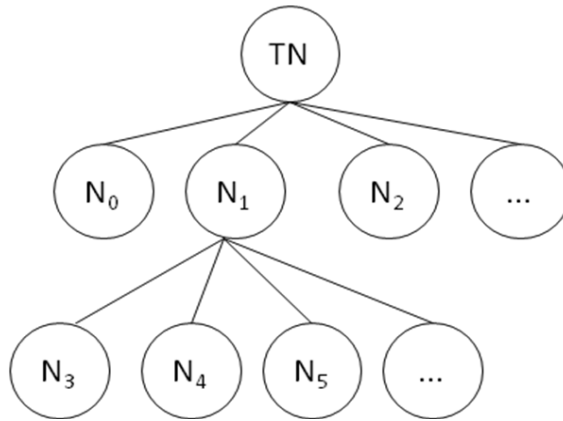
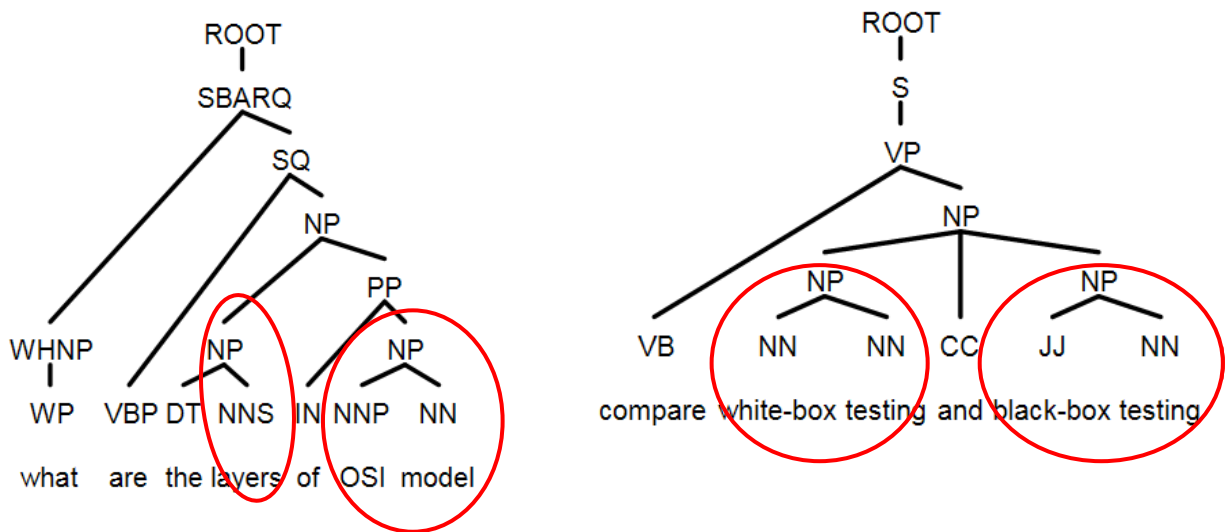


Figure 5.6: A skeleton tree used to describe the algorithm 5.1 (Dali et al., 2009)

The following two examples in Figure 5.7 illustrate the triple extraction process using different question types.

Example question 1 [type = descriptive]: *What are the layers of the OSI model?*

Example question 2 [type = comparison]: *Compare and contrast white-box testing and black-box testing.*



Triple = [OSI model, layer, ?]

Triple = [white-box testing, black-box testing, ?]

(a)

(b)

Figure 5.7: parser tree and corresponding triples for (a) question 1 (b) question 2

Triple Enhancer

Generally, the vocabulary used to write questions might not be as same as the natural language used to write lecture slides. Therefore, mapping between question triples and domain triples to extract TSKCMaps was challenging. In order to overcome this, TASF utilised an external database to identify synonyms of triple elements. Generally, synonyms for the majority of Computer Science concepts (i.e. technical vocabulary) are unavailable in lexical databases like WordNet (Miller et al., 1990). Therefore, this work utilised the SynSet library of WordNet to find synonyms of ‘verbs’ which connect two concepts. The synonym extraction algorithm using SynSet of WordNet is included in Appendix B.

5.2.4 Domain model

This component provided domain knowledge to extract task-adapted concept maps. It contained two sub components; *map merge* and *TreeGraph converter*. The domain model contained a repository of CourseCMaps (Figure 4.26) generated using CMMF (Chapter 3). The repository stored the concept maps as CXL files (Figure 4.25) (Canas, Hill, et al., 2004).

Map merge

Concept maps extracted from related topics typically leads to sharing of concepts across different concept maps (mainly due to the split of lectures between different lecture sessions). These multiple concept maps of related topics were expected to reduce the complexity of the map if they were interpreted manually.

However, when performing TSKCMap extraction, merging of related concept maps provided a richer domain model. In addition, matching question triples to extract TSKCMap from a set of related concept maps was a more complex task than considering matching within a single concept map. Figure 5.8 illustrates an example of concept map merging.

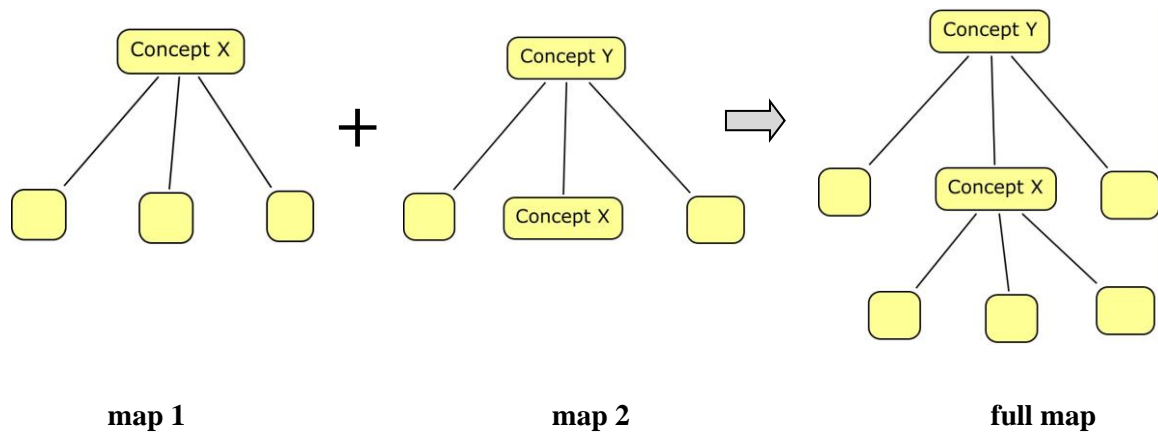


Figure 5.8: An example of concept map merging

This approach considered only the ‘root’ nodes of concept maps for merging. The algorithm first extracted the ‘root-id’ from ‘map’ element of CXL file (map 1) and searched through the ‘concept-list’ of map 2 to find the corresponding ‘concept label’. A similar approach applied for the reverse order (The algorithm of concept map merging is included in Appendix B). If both the matching nodes were roots of two maps, one map was completely merged to the other from the *root*. In other situations, only the corresponding matching parts were merged. This process also improved the cross-relations between concepts which result a rich knowledge source for TSKCMap extraction.

TreeGraph Converter

A concept map is hierarchical by nature, where each concept includes parents and children. Therefore, it can be stored in a ‘Tree’ structure for processing. Additionally, a concept map contains directed labelled edges between arbitrary pair of nodes (e.g. *non-taxonomic relations* and *cross-links*) which lead it to be mapped to a ‘graph’ structure. Therefore, a suitable data structure for storing concept maps is a graph with a tree skeleton. The Stanford Core NLP group has implemented such a data structure called ‘*TreeGraph*’ to store grammar trees (Manning et al., 2014). In the *TreeGraph* structure, concepts (or nodes) were represented as ‘*TreeGraphNode*’ and ‘*addArc*’ method connected two nodes using an edge with a label. The original implementation was slightly customised in the context of concept mapping. For instance, *TreeGraphNode* was originally implemented with a single parent. However, concept maps allowed multiple parents per node. The algorithm for converting CXL file to *TreeGraph* structure is included in Appendix B.

5.2.5 Task-adapted Concept Map Extractor

This component is the core of the *TASF*. It extracts task-adapted concept maps after matching the question triples with the *TreeGraph* structure of the *CourseCMap* (see Figure 5.4 for an example *TSKCMap*). Indices using the concept labels of *TreeGraphNode*s were constructed to match against the elements of question triple (i.e. label of the start node and end node). The ‘similarity-based exact sub graph’ matching technique is supported through a lemmatisation where labels of concepts and relations in both question triple and *CourseCMap* are mapped to their base form. Additionally, synonyms of the labels of relations are utilised. This process maintains a threshold of 15 concepts based on the suggestions by Novak and Canas (2006) and the feedback received from domain experts.

If an overlap occurred, the task-adapted concept map extraction algorithm considered the boundary of a sub graph using the features listed in Table 5.6.

Table 5. 6: Important features for task-adapted concept map extraction

Feature	Description
Question type	‘Descriptive’ or ‘comparison’
In-degree and out-degree	Number of incoming and outgoing links of each node. This determines the importance of each node
Root	Most general node of the map
Leaf nodes	Nodes without children, generally these are the most specific nodes of the map
Number of overlapping nodes	This determines the boundary of the sub graph; Overlapping nodes should always be greater than 0. If this number is greater than 1, the boundary of the sub graph needs to accommodate all the overlapping nodes and its common parents, children and siblings
Number of overlapping relations	If relations in the question triple are not overlapping with the relations in the <i>TreeGraph</i> , synonyms are considered. The overlapping of relations are not mandatory, particularly in ‘comparison’ type questions
Distance between overlapping nodes	This considers the distance in every path including cross-links

Alternatively, if an overlap did not occur using an exact matching technique, the algorithm considered triples from ‘question description’. Algorithm 5.2 illustrates the process of task-adapted concept map extraction.

Algorithm 5.2: Task-adapted concept map extraction

```
Require: TreeGraph      // TreeGraph of domain model (i.e. CourseCMap)
           Triple         // triple of question
           Question type  // 'comparative' or 'descriptive'

1. Create Hash Map from TreeGraph, label of TreeGraphNode as 'key' and
   TreeGraphNode as 'value'
2. Question type 'descriptive' requires at least one element - start node
   or end node of question triple with or without relation label
3. Question type 'comparison' requires both start and end node of question
   triple with or without relation label
4. For each non-empty elements in the question triple
   1a Match keys in the hash map
   1b If match found
       i. Obtain corresponding TreeGraphNode and create a list of
          matching nodes
   1c Else
       i. Match with each non-empty elements in the triples of
          question description
   1d End if
5. End for
6. Set the number of nodes (count) in task-adapted concept map to 15
   //threshold of the TSKCMap
   // calculate the boundary of task-adapted concept map
7. Calculate the boundary of task-adapted concept map using the features
   discussed in Table 5.6
8. Do while (count <= 15)
   1a Add TreeGraphNodes to the candidate_list according to their
       descending order of weight (ascending order of importance)
9. End Do-while

Output: task-adapted map      // list of TreeGraphNode
```

5.2.6 Visualisation

The extracted TSKCMap was represented using a *TreeGraph* structure. It is converted back to CXL format for it to be compatible with the visualisation software - IHMC CMapTools (Figure 5.4 (b)) (Canas, Hill, et al., 2004).

The TASF extracted task-adapted concept maps to provide additional scaffolding in the problem solving context. Therefore, the framework needs to be assessed with students. In order to make the TSKCMap accessible to the learners, a web-based prototype was designed and developed. The next section discusses the design consideration of the prototype.

5.3 Design of the Web-based Prototype

The design of the prototype considered several requirements. The participants of the experiments had already been familiarised with online quizzes in the Learning Management System (LMS). Therefore, the prototype was designed to provide similar look and feel with a simple design. The prototype allowed any number of attempts and provided the opportunity to practice and self-assessment. Students were allowed to skip a question only after three incorrect attempts since the intention was to assist them in learning the required skills using the task-adapted concept maps as scaffolding. The 'skip' option existed to reduce frustration. The questions constructed for the prototype were multiple choice questions (MCQ). Therefore, guessing was likely to occur which could affect the results of the experiments. In order to minimise guessing, the prototype included multiple correct answers as well as randomly shuffling answers in every second attempt.

Prior to providing scaffolding, the prototype first presented the questions. This allowed collecting students' prior knowledge. If the student lacked the required knowledge, the second attempt was presented with a link to the scaffolding. This link was a dynamic link which changes in each unsuccessful attempt (e.g. *Do you need help?*, *Struggling?*, *Need some help?*) to encourage students to utilise scaffolding. The scaffolding was shown as embedded images (treatment groups) or as PDF files (control group) with a link to go back to the quiz (see Figure 5.11 (c)).

5.4 Development of the Web-based Prototype

5.4.1 Development Environment

The prototype utilised a previously extracted TSKCMap from a pre-defined set of questions as an *off-line* process. Similar to the CMMF, the TSKCMaps were also produced as CXL files. CXL-based concept maps were converted as images (e.g. JPEG, GIF, PNG) using IHMC CMapTools to visualise in the prototype (Canas, Hill, et al., 2004).

The web-based prototype was written using PHP language (version 5.5.11) and MySQL was used as the database server (version 5.6.16) to store questions, student data and logs. The web server was Apache (version 2.4.9).

5.4.2 Interaction Design

The web-based prototype is illustrated using an interaction diagram (Figure 5.9). Interaction flow diagram shows the paths and processes that student or system will take as they progress.

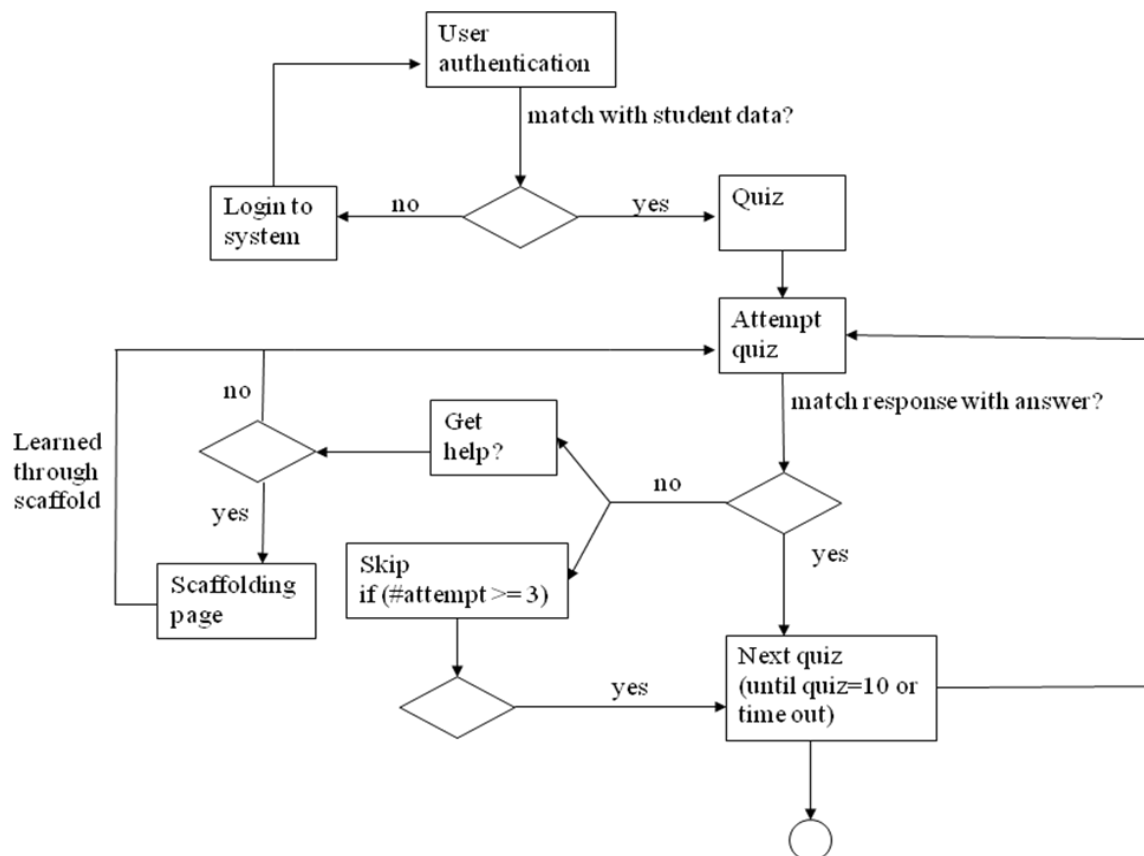


Figure 5.9: Interaction flow diagram

Each of the processes is briefly explained below.

Student login

Students were expected to log into the system using their student ID. If they successfully logged in, they could access quizzes since student IDs were previously entered to the database. Upon logging students were assigned to one of the three treatment groups or control group.

Attempt quiz

Upon successful log in, students were directed to practice quizzes. The Multiple Choice Questions (MCQ) provided the feedback about the performance immediately. Each MCQ contained five choices with a possibility of multiple correct choices. When student attempted the quiz correctly, they were forwarded to the next question until all the questions were completed.

If the attempt was unsuccessful, students were prompted to reattempt with an additional link to get 'help'. However, after three unsuccessful attempts to answer correctly, students were provided a 'skip' option to reduce student frustration. Nevertheless, they could reattempt until they obtained the correct answer.

Scaffolding

Within this stage, the student was provided with a scaffolding resource based on their experimental group. Once the student had spent adequate time to learn the necessary knowledge using the scaffolding, they could go back to the question to attempt yet again. These steps were repeated whenever student needed help and the students' interaction data, time spent and scores were recorded in the database.

5.4.3 Database Design

The quiz database contained six tables ‘*student*’, ‘*question*’, ‘*answer*’, ‘*attempt*’, ‘*reply*’ and ‘*scaffolding_attempt*’. Figure 5.10 illustrates the database design along with tables, fields, data types and their relationships with other tables.

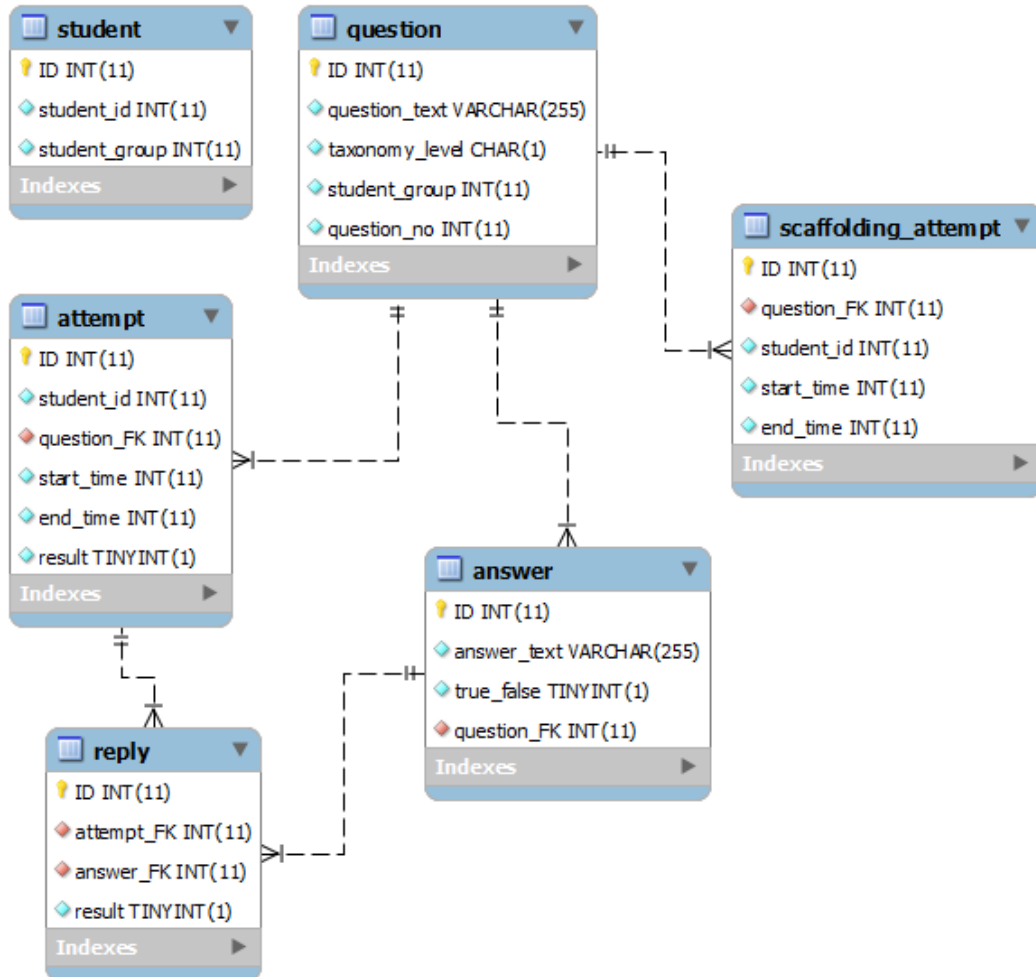


Figure 5.10: Database design for the web-based prototype

5.4.4 Web Interfaces

The web interfaces has three basic components; login screen, quiz screen and scaffolding screen as shown in the Figure 5.11.

Please enter your student ID:

Submit

(a)

Question : #1

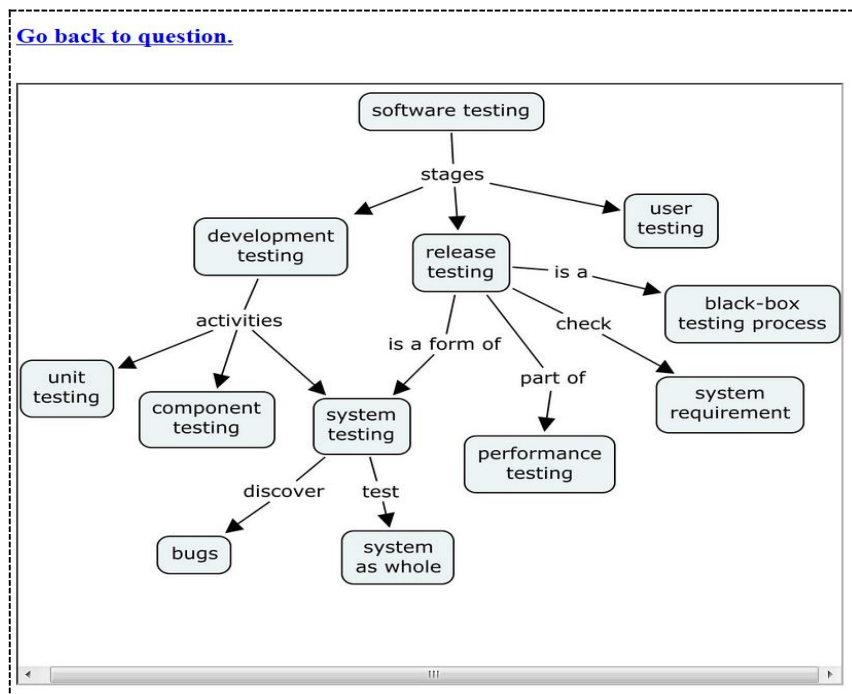
Development testing is a stage within Software Engineering life cycle which includes all testing activities carried out by the development team. Identify other testing stages

- User testing
- Regression testing
- System testing
- Acceptance testing
- Release testing

Submit

Remaining time: 18 Minutes,51 Seconds.

(b)



(c)

Figure 5.11: Screen shots of (a) user login (b) quiz (c) scaffolding resource

The scaffolding resource shown in Figure 5.11(c) is an example from the treatment group who received task-adapted concept maps. As discussed previously, other groups received different forms of scaffolding (Appendix C).

5.4.5 Quantitative Data

In every attempt of a quiz (multiple attempts are possible), the following data was collected and stored in the database for analysis.

1. The choices selected as answers and whether they were correct or incorrect
2. The time spent on each attempt
3. The time spent on each scaffolding
4. The final score for the quiz
5. The total time spent on each quiz

5.5 Summary

This chapter presented the design and development of the task-adapted scaffolding framework (TASF) and a web-based prototype. The main objective of this part of the thesis was to adopt concept maps extracted using CMMF to provide scaffolding in the problem solving context, in the case where students are lacking the required skills. This thesis is specifically focused on providing more relevant information to learning in terms of task-adapted concept maps.

The TASF consisted of four components; question processor, domain model, task-adapted concept map extractor, and visualiser. Question processor processed the input question and identified question type and question triples. Domain model consisted of concept maps extracted using CMMF. Task-adapted concept map extractor utilised question triples and question type as inputs to extract sub-graphs from CourseCMaps that overlapped with domain triples. Finally, visualiser demonstrated the extracted task-adapted concept maps with the use of IHMC CMap tools (Canas, Hill, et al., 2004).

Figure 5.12 illustrates the integration of CMMF and TASF and its overview from teachers' and students' perspective.

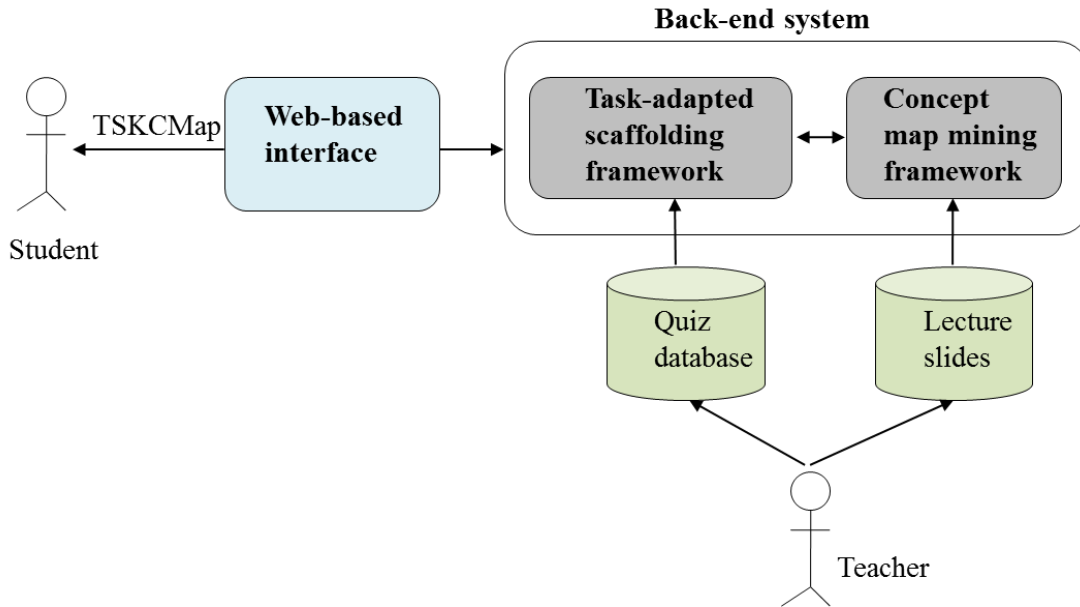


Figure 5.12: Final overview of the research

The next chapter (Chapter 6) presents the evaluation of CMMF and TASF. The first stage of the Chapter 6 presents the algorithm evaluation to measure the effectiveness of concept maps generated from lecture slides. The second stage of the Chapter 6 presents the effectiveness of task-adapted concept maps as scaffolding when compared to other forms of scaffolding resources in the context of problem solving.

Chapter 6

Evaluation

This chapter categorises the evaluation of the research into ‘algorithm evaluation’ and ‘experimental evaluation’. The algorithm evaluation measured the effectiveness of the ‘concept map mining framework’ in terms of the quality of the concept maps extracted from lecture slides. The experimental evaluation measured the effectiveness of task-adapted concept maps as scaffolding to improve the learning outcomes of students. The usefulness and effectiveness of the research was evaluated in a series of studies that focus on different aspects of the research.

The algorithm evaluation involved human experts of selected Computer Science courses (referred to as *domain experts*) as the *gold standard* to determine the quality of auto-generated concept maps using the CMMF (Ruiz-Primo & Shavelson, 1996; Villalon & Calvo, 2008). There were three main studies to evaluate the concept extraction, relation extraction and ranking algorithms. Another two supplementary studies were conducted to evaluate pronoun resolution and noise detection, to improve the performance of the CMMF. Figure 6.1 illustrates the design of studies to evaluate CMMF and TASF.

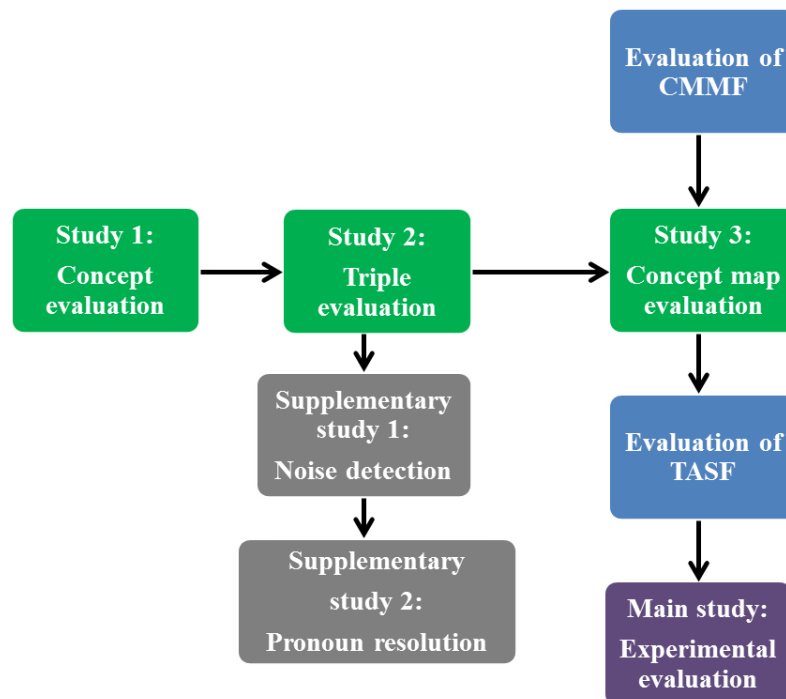


Figure 6.1: Design of the evaluation studies

The first study of algorithm evaluation tested the hypothesis '*computer-generated concepts can be used as an alternative to human extracted concepts*'. The study was carried out with six domain experts who were teaching assistants of the selected CS courses. Results revealed that concept extraction using CMMF is possible; however, the effectiveness was not as good as human extracted concepts.

The second study of algorithm evaluation tested the hypothesis '*computer-generated concept-relation-concept triples can be used as an alternative to human extracted triples*'. Two domain experts of selected courses volunteered to evaluate triples generated from CMMF. Results demonstrated that triple extraction using CMMF is promising in selected CS courses.

The final study of algorithm evaluation tested the hypothesis '*It is possible to develop auto-generated concept maps of lecture slides to strongly correlate with human constructed maps*'. This study involved an overall evaluation of CMMF as a technique to demonstrate the effectiveness of concept maps extraction from lecture slides. This study involved seven University lecturers of selected CS courses since they were the arbiters of developed lecture slides. The results revealed that the CMMF is applicable for any course within the CS domain; however, was best suited to extract concept maps from the courses classified as *well-fitted* contents.

The second stage of the evaluation measured the effectiveness of the task-adapted scaffolding framework with students. The motivation of this evaluation was to test the hypothesis of '*students who receive task-adapted concept maps as scaffolding will have an increased learning gain compared to those who did not receive task-adapted concept maps*'. In order to measure the change in learning outcome between post- and pre-tests, students were randomly assigned to a control group and three treatment groups based on the different form of scaffolding they received for problem solving. The control group received lecture slides (LS) as scaffolding while the treatment groups received different forms of concept maps. Treatment Group 1 received a full concept map (CMap) extracted using CMMF. Treatment Group 2 received the same concept map as group 1, but with the context to answer the question highlighted (HLCMap). Treatment Group 3 received a task-adapted concept map (TSKCMMap) generated using TASF.

6.1 Evaluation of the Concept Map Mining Framework

Each study of algorithm evaluation presents the study design using participants, materials used and the procedure. Afterwards, data analysis is discussed based on hypotheses developed. Finally, results are presented with discussions.

6.1.1 Evaluation Measures

Precision and *recall* were utilised for algorithm evaluation (Manning et al., 2008). Table 6.1 demonstrates the calculation of precision, recall and F-measure;

Table 6.1: Contingency table

Human	Computer	
	True	False
Positive	T_p	F_p
Negative	T_n	F_n

Precision (the fraction of retrieved documents that are relevant) = $T_p / (T_p + F_p)$

Recall (the fraction of relevant documents that are retrieved) = $T_p / (T_p + F_n)$

F-measure (harmonic mean) = $2 \times (\text{precision} * \text{recall}) / (\text{precision} + \text{recall})$

When comparing human judgement to evaluate the machine performance, the results are usually affected by the subjective and variable nature of human decisions. Therefore, the common approach is to consider and measure the agreement between evaluators which is known as *inter-rater agreement* (*inter-rater reliability*) (Manning et al., 2008).

In order to calculate the inter-rater agreement, this research utilised *positive specific agreement* proposed by Hripcsak & Rothschild (2005) over the popular *Kappa statistic* (Manning et al., 2008). The kappa statistic considers the number of concepts that neither evaluator selected. Although it is possible to identify this in document ranking in information retrieval applications, the number of potential concepts that neither evaluator selected is challenging to identify in this corpus. This idea is demonstrated in Table 6.2.

Table 6.2: Agreement between two evaluators

Evaluator 1's judgement	Evaluator 2's judgement	
	Positive	Negative
Positive	a	b
Negative	c	d

According to the Table 6.2;

a: is the number of concepts that both evaluators agree on as potential concepts

d: is the number of concepts that both evaluators agree on as not potential concepts

b and c: are the number of concepts that the evaluators disagree on

Therefore, the *positive-specific agreement* is calculated as follows;

$$\text{Positive-specific agreement} = 2a / (2a + b + c)$$

When using more than two evaluators, the inter-rater agreement is calculated using the pairwise average between all the evaluators (Manning et al., 2008).

6.1.2 Subject Domain

Since development of CMMF did not utilise any external resources, it can be applied to any discipline with availability of sufficiently mineable lecture slides. However, this thesis demonstrated CMMF using the Computer Science (CS) domain.

The corpus of lecture slides included 27 core Computer Science courses and 3 courses obtained from text book publishers (Silberchatz et al., 2012; Sommerville, 2010; Stalling, 2007) (Appendix A). Each course contained approximately 22 topics, which results an analysis of a collection of nearly 600 slide sets. Additionally, each slide set contained approximately 40 slides. Therefore, the CMMF analysed nearly a corpus of 24,000 slides to propose new techniques to extract useful knowledge as concept maps from lecture slides.

6.1.3 Evaluation of Concept Extraction

The objective of this study was to answer the research question;

‘Can computer-generated concepts be used as an alternative to human extracted concepts?’

Study Design

In order to evaluate the concept extraction algorithm, two Computer Science courses (*Software Engineering* and *Computer Networking*) each containing five slide sets were used (see Appendix A). Each slide set contained approximately 50 slides resulting 250 slides per an evaluator. The slide sets contained combinations of text, figures, examples, case studies and programming codes.

This study involved six teaching assistants from the Computer Science School as independent evaluators. They were neither involved in nor aware of the underlying techniques used in the

research. Three of them agreed to evaluate *Software Engineering* course and used the same slide set while others evaluated *Computer Networking* slide set since they had prior knowledge and experience in teaching, tutoring or practical supervision of these courses. Additionally, everyone had completed these courses with high grades in their Undergraduate studies. As discussed in the ‘CMM review’ (Section 3.3), the involvement of three human evaluators is common among other related works (N.-S. Chen et al., 2008; Leake et al., 2004; Villalon & Calvo, 2011).

The task of the participants was to identify important domain concepts in the given lecture slides. They were requested to underline the concepts that they thought were important to the given domain based on their knowledge and experience. The task was completed either on computer or on paper. Most participants completed it on the computer.

Although these evaluators had expertise in the subject matter, some of them had difficulties in identifying what was a correct concept and how to determine it. An example of feedback received from an evaluator during the first cycle of annotation is listed below.

“In several cases, I could not distinguish between a ‘concept’ and a ‘property/feature/capability’ of the presented item, this is why in some cases a lot is marked, and then some other slides are empty”

Clues were provided to assist evaluators in identifying domain concepts (e.g. CS concepts might be identified as the relevant terminology of the domain). Further, a sample slide and examples of concepts using a different domain were given to them. However, they were not provided with clues that concepts might be nouns or noun phrases due to two reasons. First, the underlying algorithm used this as a hypothesis. This prevented any influence arising from selecting all nouns and compound nouns as domain concepts. Conversely, CS domain contains verbs, adverbs and adjectives as important concepts. For instance, some verbs in its *-ing* form can be utilised as nouns (*gerund* verbs).

Data Analysis

The underlined domain concepts were lemmatised to map them to their base form and then extracted as a list of concepts (Manning et al., 2014). This list was kept as the reference model to compare with the machine-extracted concepts. From the human annotated data, some concepts were eliminated from the evaluation since they were located within figures and tables. CMMF did not extract information from figures and tables. This issue will be discussed as a ‘limitation’ of CMMF.

The performance of concept extraction algorithm was obtained using *precision*, *recall* and *F-measure*. Since, this study used three human evaluators to annotate a single slide set, the average of pairwise inter-rater agreement was calculated between three evaluators.

Results and Discussion

The total number of concepts retrieved using computer from all the 10 slide sets was 1210. Since there were approximately 50 slides in each slide set, the average number of computer extracted concepts per slide were 2.4. The total number of concepts extracted by six human evaluators was 1800. Therefore, the average number of concepts identified using human evaluators per slide was 3.6. Due to the ranking mechanism incorporated with the concept extraction algorithm, it was anticipated that the computer retrieved concepts would be fewer than human annotations.

Table 6.3 summarises the results for Software Engineering and Computer Networks using *precision*, *recall* and *F-measure* based on each evaluator. In the final row, the average for each course is presented.

Table 6.3: Evaluation results of concept extraction

Evaluator	Software Engineering			Computer Networks		
	Precision	Recall	F-measure	Precision	Recall	F-measure
E1	0.304	0.534	0.386	0.277	0.554	0.358
E2	0.422	0.526	0.465	0.366	0.592	0.452
E3	0.484	0.626	0.541	0.283	0.600	0.382
Average	0.403	0.562	0.464	0.308	0.582	0.397

The average performance for the Software Engineering course was 40%, 56% and 46% for *precision*, *recall* and *F-measure* respectively. The Computer Network course obtained average precision, recall and F-measure of 31%, 58% and 40% respectively. On average this was 43% of *F-measure* (human-to-machine agreement) for concept map extraction algorithm.

The comparison between the results obtained from this evaluation and other existing works in the concept map mining area was not achievable since there are no prominent existing works which extract concept maps from *lecture slides*. Some preliminary works such as Ono et al. (2011) achieved an overall performance (similar to the *precision*) of 28% and Gantayat & Iyer (2011) reported performance (similar to the *F-measure*) of 40%.

Therefore, an alternative approach was used to evaluate the results. According to the notion proposed by Hearst (2000), the results obtained from computer algorithms are effective, if they

are greater than or equal to inter-rater agreement. Therefore, the inter-rater agreement was calculated between each pair of evaluators and obtained the average since there were three evaluators per each course.

Table 6.4: Comparison between F-measure and inter-rater agreement; SE - Software Engineering, CN - Computer Networks

Lecture	SE1	SE2	SE3	SE4	SE5	CN1	CN2	CN3	CN4	CN5
F-measure	0.464	0.604	0.419	0.445	0.386	0.476	0.431	0.452	0.347	0.311
Inter-rater agreement	0.518	0.598	0.540	0.54	0.516	0.428	0.454	0.465	0.353	0.388

According to the results in the Table 6.4, it is evident that only two courses out of 10 achieved F-measure greater than inter-rater agreement. The overall F-measure was 43% while inter-rater agreement was 48%. This result does not agree with the notion proposed by Hearst (2000).

As discussed in Chapter 3, the inter-rater agreement of 80% is taken as ‘good’ agreement and values between 67% and 80% are taken as ‘fair’ agreement. When the agreement is below 67%, the corpus under study believed to be uncertain for human annotation (Manning et al., 2008). Since, the overall inter-rater agreement of this study was below than the ‘fair’ agreement (48%), the comparison between human-to-machine agreement and inter-rater agreement is not encouraged due to the uncertain nature of corpus under study.

The first five slide sets were obtained from the Software Engineering course. These lecture slides were constructed with the use of text book slides (Sommerville, 2010). Therefore, these lecture slides contained relatively meaningful text and grammatically complete sentences or phrases, good hierarchy and summarisation. The three evaluators who annotated these slide sets had relatively higher agreement around 0.55 since the potential concepts have a clear separation from the redundant data. The last five slide sets were obtained from the Computer Networking course, which largely contained brief phrases (Stalling, 2007). Therefore, some evaluators considered the entire slide as ‘important’ without considering any of the individual domain concepts. This resulted in a poor inter-rater agreement (0.42) between evaluators. This occurred repeatedly throughout their evaluations, indicating coarseness in granularity when using human involvement. In addition, each evaluator had to deal with nearly 250 slides, causing them to miss some important concepts in the review. These issues emphasised the error-prone and variable nature of manual knowledge extraction, stressing the importance of machine-based extraction.

In contrast to the issues with the inter-rater agreement, the performance of the concept extraction algorithm required further improvements such as pronoun resolution and a

mechanism to identify concepts with their corresponding prepositions which should improve the human-to-machine agreement. Additionally, the results suggested the necessity of an automatic noise detection mechanism to eliminate redundant data. Thus, the next series of evaluation studies addressed each of these issues.

6.1.4 Evaluation of Noise Detection

CMMF developed an automated noise detection technique using co-occurrence analysis between important domain-specific contents and redundant data. In order to apply this algorithm to CMMF for noise elimination, it is important to evaluate it separately since accuracy of this algorithm can affect the final results of the CMMF performance.

Study Design

A random collection of 27 slide sets was selected from Computer Science courses included in a courseware repository of Computer Science School and materials obtained from text book publishers (see Appendix A). A slide set consisted of approximately 40 slides. The selected materials contained a combination of domain-specific and redundant data.

This study involved a single independent evaluator. Noise detection was not a highly subjective decision like concept identification since redundant data was ‘obvious’ to identify.

The task of the evaluator was to identify which slide contents were inappropriate for inclusion in a concept map and then annotate the corresponding slide numbers. This was a relatively straightforward process. This study requested the evaluator to ‘flag’ any doubtful situations. Prior to analysing the data, a discussion has been carried out with the evaluator to clarify the conflicts. In situations where conflicts could not be resolved through discussions, such slides were manually eliminated from the analysis.

Data analysis

The basic human to computer comparison was carried out to obtain *precision* and *recall*. The hypothesis for this study was ‘*if the topic (i.e. root) of the course or title of a slide does not co-occur fully or partially (as word tokens) with the content of the slide, then the whole slide can be eliminated as unrelated*’.

The process of evaluation did not specifically mention anything about figures or tables included in lecture slides, even though CMMF did not extract information from figures and tables. Therefore, the evaluator had included the slides that contain important figures and tables as ‘relevant’ due to two reasons. First, they might think that the underlying algorithm will extract useful knowledge from figures or tables. On the other hand, some concept maps attach figures

as resources. Therefore, they excluded such slides in the annotation list (*note*: annotation list included only ‘redundant’ list of slides). However, inclusion of some figures or tables, particularly with category headings (e.g. *example, case study*) was not suitable for a concept map.

Therefore, the noise detection algorithm was evaluated in two stages. The first stage compared the human annotations with computer and presented the results as ‘*normal performance*’ in Table 6.5. The second stage conducted another evaluation on the same data by eliminating slides which had ‘figures’ and ‘tables’ issues and the results of this stage was presented as ‘*edited performance*’ in Table 6.5.

Results and Discussion

This study analysed nearly 1100 slides (i.e. 27 slide sets with approximately 40 slides per slide set) to identify redundant slide data. This was compared to human annotations and the results of a sample of randomly selected slide sets are presented in Table 6.5.

Table 6.5: Evaluation results of noise detection

Lecture (Sample slide sets)	Normal performance			Edited performance		
	Precision	Recall	F-measure	Precision	Recall	F-measure
1	1	1	1	1	1	1
2	1	0.714	0.833	1	0.714	0.833
3	0.8	1	0.889	1	1	1
4	0.75	1	0.857	0.75	1	0.857
5	0.6	0.75	0.667	1	1	1
Average (sample of 5 slide sets)	0.83	0.892	0.849	0.95	0.943	0.938
Average (total of 27 slide sets)	0.85	0.898	0.873	0.938	0.964	0.950

According to the average results for 27 slide sets, it was evident that the ‘normal performance’ stage 1 (i.e. stage 1) achieved an average of 85%, 90% and 87% for *precision*, *recall* and *F-measure* respectively. In this stage, the evaluator might consider the figures and tables with category headings as ‘relevant’. The second stage fixed some issues with the consultation of the evaluator to obtain ‘edited performance’. Therefore, second stage had a performance improvement to 94%, 96% and 95% for *precision*, *recall* and *F-measure* respectively. Regardless of the figures and tables issue, both the stages achieved promising results.

Even though the algorithm was evaluated using two stages, the original noise detection of CMMF is similar to the scenario in ‘edited performance’ since the underlying mechanism of the algorithm eliminated the figures and tables captioned using category headings. Such slides did not provide adequate information to decide whether the content is domain-specific or redundant.

6.1.5 Evaluation of Pronoun Resolution

This research developed a pronoun resolution algorithm specific to lecture slides. The objective of this study was to investigate the research question;

‘Can computer-based replacements of pronouns be used as an alternative to human identified replacements of pronouns?’

Study Design

In order to evaluate the pronoun resolution algorithm, fifteen computer science slide sets were selected. These slide sets include courses taught across different undergraduate levels at University of Adelaide such as *Introductory Programming* and *Object Oriented Programming* (level 1), *Computer Science Concepts*, *Software Engineering* and *Database and Information Systems* (level 2), *Computer Architecture*, *Operating Systems*, *Computer Networking* and *Distributed Systems* (level 3), and *Software Architecture* and *Software Process Improvement* (level 4) (see Appendix A). Each slide set contained approximately 40 slides. Each of these courses contained combinations of different writing styles.

Two independent evaluators were recruited who had knowledge and expertise in Computer Science concepts. They were native and fluent English speakers who could identify pronouns and their corresponding replacement candidates in the given context.

The task of evaluators was to find the most suitable replacement candidates for pronouns contained in lecture slides. A list of candidate replacements extracted by the system was not provided to the evaluators since it could influence judgement of which replacement was most suitable. Evaluators were requested to leave ‘blank’ or flag when they could not find replacement candidates from the context of a slide.

Data Analysis

The replacement candidates identified by the human experts were compared with computer predicted replacements. The data collected from pronoun resolution task included ‘blank’ or flagged records in both lists (i.e. human and computer). When both the lists flagged the same pronoun replacement, that data was removed from the evaluation. However, when one list contained a flagged data cell, those data was considered for the analysis.

Similar to study 1, this evaluation used *precision*, *recall* and *F-measure* to measure the human-to-machine agreement and calculated the inter-rater agreement to compare the agreement between human evaluators.

Results and Discussion

The statistics about pronouns under study is presented in Table 6.6.

Table 6.6: Statistics of pronouns discovered in the corpus

Pronoun	they	their	it(s)	itself	we	them	you(r)	us
Frequency	57	51	241	17	23	34	94	22

The pronouns discovered in the corpus included *you*, *we*, *us*, *itself* to address students who referred to the course material or listened to the lecture (e.g. *Development testing includes unit testing in which you test individual objects*). Due to the lack of named entities to identify replacement candidates for ‘gender-related’ pronouns in the Computer Science domain, this research excluded the pronouns - *us*, *you*, *we* and *itself*. Therefore, a total of 383 pronouns were resolved. Table 6.7 illustrates the results of the F-measure and inter-rater agreement.

Table 6.7: Evaluation results of pronoun resolution

Lecture	1	2	3	4	5	6	7	8
Frequency	17	16	0	67	54	39	44	50
F-measure	0.857	0.66	-	0.746	0.923	0.587	0.5	0.571
Inter-rater agreement	0.8	0.33	-	0.916	0.9	0.727	0.8	0.68

Lecture	9	10	11	12	13	14	15
Frequency	5	10	7	7	40	23	4
F-measure	0.5	0.909	0.19	0.41	0.528	0.857	0.66
Inter-rater agreement	0.6	1	0.142	0.571	0.384	1	0.5

Table 6.7 provides evidence that the use of pronouns in lecture slides varies dependent on the author. For instance, L3, which was derived from the *Computer networks* book slides (Stalling, 2007), provided grammatically complete sentences or fragments without use of pronouns. It is evident that courses which demonstrated grammatically rich, consistent writing styles provided probable replacement candidates within the context, allowing the computer algorithm to accurately (F-measure > 80%) perform pronoun resolution (e.g. L1 – *Software Architecture*, L5 – *Operating System*, L10 – *High Integrity Software Engineering*).

The pronoun resolution algorithm achieved an average F-measure of 64%. However, the comparison of this value with related works is not consistent since the corpus under study (i.e. lecture slides) is different from technical manuals used by Mitkov (1998) or the DUC data set used by Leskovec et al. (2004).

As highlighted in Table 6.7, six courses demonstrated that the F-measure was greater than or equal to inter-rater agreement. This supported the hypothesis for several courses. The average

inter-rater agreement was 67% which is considered as a ‘fair’ agreement between evaluators (Manning et al., 2008).

Human agreement was substantially reduced when one rater suggested a replacement candidate while the other flagged it when they found it uncertain. Occasionally, some of machine replacements did not overlap with human, reducing the accuracy as shown in L11 and L12. These generally included more than one dependent clause which made it difficult for even human interpretation (e.g. *need to write it to disk before replacing it if it was updated since it was last fetched from/written to disk*). In addition, some sentences included ‘dummy’ (exophoric) pronouns which did not contain a corresponding replacement in the context. Therefore, ‘dummy’ pronouns were usually excluded from processing (N. Ge, Hale, & Charniak, 1998) For instance, in the sentence ‘*it is raining*’, the pronoun ‘*it*’ adds no meaning to the sentence but it was required by syntax.

The next section describes the evaluation of triple extraction. This study utilises the same set of CS courses as in pronoun resolution since both of these studies performed together. However, different slide sets (i.e. topics) were obtained.

6.1.6 Evaluation of Triple Extraction

In order to measure the effectiveness of triple extraction algorithm, a study was carried using human experts as the *gold standard*. The objective of this study was to investigate the research question;

‘Can computer-generated concept-relation-concept triples be used as an alternative to human extracted triples?’

Study Design

This study utilised the same CS courses used in the evaluation of pronoun resolution (see Section 6.1.5). However, the slide sets (i.e. topics) used for this study were different from the pronoun resolution to ensure the experiments were conducted across a diverse range of corpus.

The same independent evaluators who worked on pronoun resolution were involved in this study. They were knowledgeable to identify subject-verb-object (SVO) from English sentences. The identification of SVO triples from *simple sentences* was straightforward while the evaluators spent a longer time to decide the triples in *complex sentences*.

The task of each evaluator was to identify subjects, verbs and objects from the given English sentences. The majority of the sentences considered were simple to annotate. One of the limitations identified in the concept extraction study (i.e. Study 1) was the poor agreement

between human and computer to combine prepositions with the corresponding concepts. Therefore, this study requested evaluators to consider incorporating prepositions where possible. The extremely complex sentences whose triples were difficult to identify manually were allowed to be flagged.

Data Analysis

Although evaluators were requested to use exact wordings in the lecture slides to annotate triples, there was no guarantee that human annotation would be identical to machine-extracted triples. Therefore, the string similarity between each subject, verb and object was calculated, providing the average score for a triple. An example of similarity calculation between subjects is shown below (Dali & Fortuna, 2008; Park & Calvo, 2008).

- **Computer** (subject) – *application programmers' interface between process*
Number of tokens (subject) - 5
- **Human** (subject`) – *application programmer interface*
Number of tokens (subject`) – 3

$$\begin{aligned}\text{Similarity (subject, subject`)} &= \text{overlap} / (\text{number of tokens in } x; x = \max(\text{subject, subject`})) \\ &= 3/5 = 0.6\end{aligned}$$

Verb and object similarity was calculated using the same steps and the average between computer and human was obtained. Therefore, the final similarity between computer and human ranged between 0-1, where 1 meant identical and 0 meant no overlap.

Similar to the pronoun resolution study (Section 6.1.5), when both lists (i.e. human and computer) flagged the difficult sentences to process, those data were removed as 'redundant'. There were other data points where elements of triples were partially extracted (i.e. 1 or 2 elements). Although missing verbs between subjects and objects are acceptable as shown in Figure 4.23, the missing subject or object that occurred in both lists were also eliminated. However, if this issue occurred in just one list (i.e. either computer or human), such triples were considered for the evaluation. The analysis removed 265 sentences as 'redundant' since both computer and human could not identify triples from them.

Results and Discussion

1996 sentences were extracted from 15 slide sets with approximately 40 slides per slide set. The average number of sentences per slide was 3.3. From the collection, 265 sentences were eliminated due to redundancy such as their extreme complexity or inclusion of 'insolvable' pronouns (see Table 6.6). A collection of 1838 triples were extracted from the rest of the 1731 sentences. A sentence could consist of no, one or more triples.

Precision and *recall* were used to calculate the F-measure (human-to-machine agreement). Similar to the previous studies, the *inter-rater agreement* between two human evaluators was obtained to overcome the subjectivity of human judgement.

Table 6.8: Comparison between F-measure and inter-rater agreement

Lecture	1	2	3	4	5	6	7	8
No. of Triples	107	81	24	173	207	108	145	221
F-measure	0.862	0.507	1	0.605	0.872	0.397	0.88	0.944
Inter-rater agreement	0.928	0.808	1	0.761	0.930	0.623	0.88	0.975

Lecture	9	10	11	12	13	14	15
No. of Triples	82	134	74	180	130	110	62
F-measure	0.787	0.833	0.465	0.319	0.497	0.858	0.672
Inter-rate agreement	0.829	0.792	0.66	0.645	0.72	0.844	0.76

According to Table 6.8, it is evident that seven courses produced acceptable machine performance (F-measure > 80%) including L1 (Software Architecture), L5 (Operating System), L8 (Software Engineering) and L14 (Software Process Improvement). Computer Networking materials (L3) prepared from the text book slides (Stalling, 2007) achieved the maximum F-measure possible (F-measure = 100%), resulting in an ideal machine extraction mimicking a human expert. The source of this course was well-written using simple English sentences or fragments (Stalling, 2007). In general, these courses included rich grammar, complete sentences with apparent independent clauses, infrequent use of complex sentences or confusing idioms. The algorithm observed to be more effective (F-measure > 80%) for courses categorised as *Software Engineering, Computer Architecture, Communications and Security* (see the subfields defined by ACM classification in (ACM, 2012)). Therefore, these contents were categorised as *well-fitted* for CMMF.

Other courses which included a combination of relatively good text contents and notations, such as *Databases* and *Distributed Systems* were categorised as *average-fitted* contents (F-measure between 50% and 80%).

Finally, the courses which demonstrated low F-measure (< 50%) were identified as *Programming languages and Compilers, Algorithms and Data structures, Mathematical foundations* and were classified as *ill-fitted* contents.

This classification will be further supported using a different analysis in the next study under ‘evaluation of concept map extraction’.

In addition, results illustrated that the human-to-machine agreement (i.e. F-measure) was greater than or equal to inter-rater agreement in four courses (highlighted in Table 6.8). This supported

the hypothesis for several courses. The average F-measure of all courses was 69%, while inter-rater agreement was 81% (> F-measure). It is practically challenging for machine to outperform human annotation in a corpus like lecture slides. Although creating lecture slides follows a consistent presentation template, there is no well-defined structure for writing course materials and it is free-form depending on the writing and presentation style of the lecturer. Therefore, the overall performance of a knowledge source like lecture slides is difficult to assess since contents of the lecture slides are not consistent like text books (Olney, 2010; Olney et al., 2011) or academic articles (N.-S. Chen et al., 2008).

The agreement between computer and human varied when one list included prepositions, adjectives and verbs to modify triple elements while the other list contained only the exact words without modifiers. The machine performance was substantially reduced in some occasions (e.g. L6, L11 and L12). This occurred mainly due to the failure of CMMF to handle negations correctly. Lecture slides occasionally include negations to emphasise pros and cons of fundamental CS concepts, and bad programming habits. The failure to correctly recognise negations led to produce completely wrong triples (e.g. *the class has no member function or member variable*). Thus, a primary effect led to a poor F-measure occurred due to the natural skill of human detecting negations.

When compared to the performance of concept extraction (F-measure was 43%) in Study 1, the improvements in the triple extraction algorithm led to an increased F-measure of 69%.

The study discussed here was restricted to determine the accuracy of subject-verb-object triples of English sentences. However, the pedagogical importance of these triples to construct concept map will be discussed in the next study.

6.1.7 Evaluation of Concept Map Extraction

Previous studies of algorithm evaluation focussed on evaluating the automated extraction of individual elements such as concepts, triples and pronouns of lecture slides. This final study put them together and evaluated the generated concept maps as a whole. Therefore, this study involved the lecturers of the corresponding Computer Science courses.

The objective of this study was to answer the research question;

‘Can computer-generated concept maps be used as an alternative to human extracted concept maps?’

Study Design

This study proposed three candidate ranking models to compare with domain experts as *gold standard* to determine which ranking model was best fitted to evaluate overall concept maps. Among them, the third ranking model (i.e. *structural feature model*) utilised several structural and graph-based features to determine the importance of concepts. Weights were allocated for each structural and graph-based feature to determine the influence of each feature for ranking. Therefore, as discussed under the ‘review of CMM’ in Section 3.2, the adjustment of weights was carried out on a *training set* and applied the adjusted weighting function to the *test set* (Manning et al., 2008).

The training set utilised previously annotated slide sets for the ‘evaluation of concept extraction’ (Atapattu et al., 2012). The previous study involved three evaluators to identify domain concepts from five Computer Networking slide sets containing 250 slides. Therefore, the concepts that were agreed by all the three evaluators were selected as ‘*most important*’ and removed from the list. From the remaining list, the concepts that were rated by any two of the evaluators were selected as ‘*important*’. Finally, the rest of domain concepts rated by a single evaluator were selected as ‘*least important*’.

For the test set, seven other Computer Science courses across different Undergraduate levels were selected. The seven courses chosen were *Introductory Programming (IP)*, *Algorithm Design and Data Structures (ADDS)*, *Object Oriented Programming (OOP)* (level 1); *Software Engineering (SE)* (level 2); *Distributed Systems (DS)*, *Operating Systems (OS)* (level 3); and *Software Architecture (SA)* (level 4) (An outline of these courses included in Appendix A).

Seven lecturers who have extensive experience in teaching CS courses from the Computer Science School volunteered to assist with the experiments. This study required participants to rate the domain concepts according to their importance. The importance of knowledge

components of a domain depends on various factors such as learning objectives, graduate attributes, examination or assessment perspectives and the attitudes of lecturers. Therefore, this study incorporated lecturers as the domain experts since they are the arbiters of their own content. The judgement was expected to reflect personal opinions based on their knowledge and perception. However, tips were provided, such as how the importance of a concept can be affected by the learning outcome, course objective, and examination perspective. These instructions did not have any relation with the factors considered in developing ranking models.

Colour pens and printed lecture slides were provided to the participants who preferred working in a paper-based environment. The rest used their computers or tablets to highlight the domain concepts. The three rating scale given to the participants consisted of 'most important', 'important', and 'least important' using three colours 'red', 'yellow' and 'green' respectively. Participants tended to rate single concepts as well as noun phrases.

During the experiments, the machine-extracted concept maps were not shown to the participants. They only had access to the lecture slides. This prevented any influence arising from structure or layout of concept maps to the judgements of evaluators.

Data Analysis

A simple computer program was utilised to extract the annotations of evaluators. The highlighted human annotations were categorised and sorted based on their ranks from 3 to 1 (*most important to least important*). Similarly, the CMMF arranged important concepts according to the ranks assigned by each candidate models.

In order to compare the human rankings with computer assigned ranks, a statistical measure (*Spearman correlation coefficient - r_s*) to compare ranking correlation was used.

$$r_s = (1 - 6 \sum d_i^2) / (n(n^2 - 1)); \quad d = \text{difference between ranks, } n = \text{sample size}$$

The hypotheses for each ranking model are listed below.

Hypothesis 1 (Baseline model): *Text location allocated by the natural layout of the presentation slides might influence human judgement of which concepts are most important.*

Hypothesis II (Linguistic feature model): *Simple grammatical structures (nouns, noun phrases) of lecture slides might have higher influence than complex grammatical structures (nested sentences, dependent clauses, indirect objects) on human judgement of which concepts are most important.*

Hypothesis III (Structural feature model): *Structural (degree centrality, proximity) and distributional (term frequency, degree of co-occurrence, typographic information) features might influence the human judgement of which concepts are most important.*

Results and Discussion

This study extracted 678 concepts from 378 lecture slides. The average number of concepts per slide was 2.2 except in *Introductory programming (IP)* course. In IP, multiple slides repeated the same content using animations. Therefore, in IP, the average number of concepts per slide was 0.8.

The base line model allocated rank ‘3’, ‘2’ and ‘1’ for text located in *titles, bullet-points* and *sub-points* respectively (see Table 4.13) and rank ‘0’ for the concepts annotated by human, but not retrieved by the machine. Similarly, experts ranked ‘3’ as ‘*most important*’, ‘2’ as ‘*important*’, ‘1’ as ‘*least important*’ and ‘0’ for the concepts retrieved by the computer algorithm, but not considered by the human expert. The two ranking lists were compared using ranking correlation coefficient and results are presented in Table 6.9.

The results are interpreted as strong positive or strong negative if r_s is close to +1 or -1 respectively. There is no linear correlation when r_s is close to 0 and the variables are considered to be independent.

Table 6.9: Spearman’s ranking correlation (r_s) of baseline model

CS Course	r_s
Software Engineering (SE)	0.193
Algorithm design and data structures (ADDS)	0.436
Introductory programming (IP)	0.113
Operating systems (OS)	0.325
Distributed systems (DS)	0.183
Object-oriented programming (OOP)	0.287
Software architecture (SA)	0.605

The correlation (r_s) is close to 0 for the majority of the courses (highlighted in Table 6.9). This demonstrated that there was no linear correlation between human rankings and the concept ‘location’ allocated by the natural layout of the presentation framework. This causes to question and reject the hypothesis I that assumed ‘*most important, important and least important concepts are located in titles, bullet points and sub points respectively*’. This result claims that the previous work by Kinchin (2006a) which performed ‘topic extraction’ should focus on finer granularity of text analysis in addition to the lecture headings. The feedback obtained from

lecturers' judgement is important for students. This implies that the layout of the slides is not overlapping with lecturers' judgement of what is more important in the lecture.

However, if this research could expand the ranking to a few other levels, a slightly more positive correlation could be expected from the baseline model. This occurred when the ranking model categorised remaining concepts as false positive (rank = 0), i.e. those that have not been ranked by human, and false negative (rank = 0) for that have not been retrieved by machine, but annotated by human.

The linguistic feature model assumed that the grammatical structure of text (noun or noun phrases, simple sentences, and complex sentences) has an impact to determine the importance of concepts. Similar to the baseline model, this model assigned higher rank (rank = 3) for noun or noun phrases, medium rank (rank = 2) for simple grammatical structures (simple sentences) and lower rank (rank = 1) for complex grammatical structures (complex sentences) (see Table 4.14). The results in the Table 6.10 show the correlation is closer to 0 for all the selected courses. This reveals that, in addition to single terms and brief phrases, simple and complex sentences might contain important domain concepts. Therefore, a finer granularity in text analysis regardless of their grammatical complexity is important to extract the useful knowledge from lecture slides.

Table 6.10: Spearman's ranking correlation (r_s) of linguistic feature model

CS Course	r_s
Software Engineering (SE)	0.247
Algorithm design and data structures (ADDS)	0.252
Introductory programming (IP)	0.293
Operating systems (OS)	0.240
Distributed systems (DS)	0.129
Object-oriented programming (OOP)	0.347
Software architecture (SA)	0.050

As discussed in the study design, the structural feature model involved training the algorithm first to adjust the parameter values. The weighting function was trained using previously annotated data for the concept extraction study (Atapattu, Falkner and Falkner, 2012). The influence of each feature (discussed in Section 4.2.5) was determined by the parameter values (Table 6.11). For instance, concepts with high out-degree can be more general, thus more important than terms with high in-degree. Weights of each feature were normalised within the range of 0-1.

Table 6.11: Best fit parameter values for structural features

Feature	Best fit parameter values
Out-degree	0.923
Proximity	0.853
Typographic information	0.764
Co-occurrence	0.559
Frequency	0.514
In-degree	0.281

After obtaining the best fit parameter values using the training set, an aggregate weight for each concept in the test set was calculated and sorted the concepts in the descending order of weights. The system defined *upper*, *medium* and *lower* threshold values in order to rank the *most important* (above upper), *important* (in-between upper and medium) and *least important* (in-between medium and lower) domain concepts. These three threshold values varied depending on the number of concepts retrieved in each slide set. Finally, similar to other two candidate models, the ranks given by participants were compared with machine prediction. The results are demonstrated in Table 6.12.

Table 6.12: Spearman's ranking correlation (r_s) of structural feature model

CS courses	r_s
Software Engineering (SE)	0.805
Algorithm design and data structures (ADDS)	0.435
Introductory programming (IP)	0.353
Operating systems (OS)	0.715
Distributed systems (DS)	0.455
Object-oriented programming (OOP)	0.521
Software architecture (SA)	0.806

Two contradictory ideas arise when this research ranks and filters the concepts above a threshold value. First, according to the definition of Novak & Gowin (1984), 15 to 25 concepts are adequate for a concept map. This involves obtaining smaller number of triples with the aim of limiting the number of concepts in a concept map to a range of 15 to 25. Conversely, the task-adapted scaffolding framework discussed in Chapter 5 required a relatively large number of important triples to produce a rich domain model, allowing extraction of more accurate task-adapted concept maps. Therefore, the ranking mechanism in this research had no intention to reduce the concepts to a range of 15 to 25 concepts. The size of the concept maps (in terms of number of concepts) varied depending on the number of extracted triples, number of triples above the threshold value and number of slides in the source document. For instance, a slide set with more than 40 slides was difficult to summarise into a range of 15 to 25 concepts since an average number of concepts per slide as determined by domain experts was 2.2.

The results of Table 6.12 show satisfactory correlation ($r_s > 0.5$) for the majority of the courses and strong positive correlation ($r_s > 0.7$) for SE, SA and OS courses. For instance, in a topic of *Software Engineering*, 55% of concepts (out of 64) overlapped between computer and human (distance = 0) and 39% of concepts indicated one level difference between ranks (Figure 6.2). This implied 94% of concepts extracted from machine algorithms were closely aligned with human judgement, resulting in a machine extraction of approximate expert maps ($r_s = 0.813$).

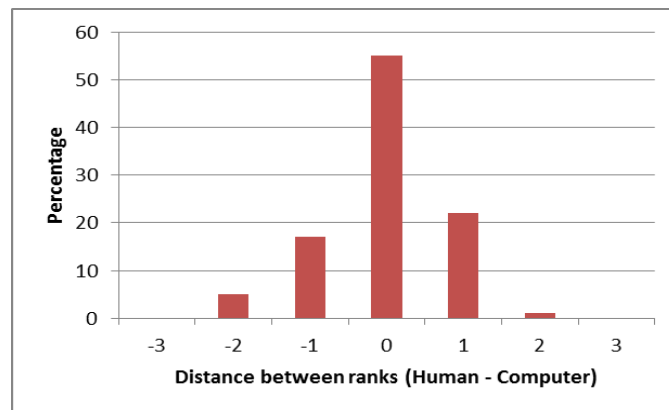


Figure 6.2: Distance between human and computer ranks against percentage of concepts ($r_s = 0.813$)

Both OS and SE slide sets were constructed using slides from text book publishers (Silberchatz et al., 2012; Sommerville, 2010) and the SA slide sets were well-written and structured. Therefore, those topics contained aspects such as rich grammar, less ambiguity, good summarisation and emphasis of important domain concepts. The *well-fitted* contents assisted successful machine extraction.

Based on the classification of previous study (Section 6.1.6), the data ranged from *well-fitted* (e.g. SE and SA) to *ill-fitted* (e.g. IP and ADDS) contents for ‘machine interpretation’.

Similar to the previous classification (Section 6.1.6), courses which included a combination of relatively good text contents and notations (e.g. DS) with $r_s \sim 0.5$ were categorised as *average-fitted* content. Although, programming courses fitted into the *ill-fitted* category for CMMF, the results from this study provided an exception for *Object oriented programming* with a correlation coefficient of 0.521. This occurred since the randomly selected lecture topic of OOP (i.e. *memory*) contained more facts than programming components.

Conversely, the remaining course topics, included combinations of category headings (e.g. *review, summary, welcome*), additional text boxes with excessive text content, ambiguous terms that were difficult to resolve such as programming and mathematical notations, were classified

as *ill-fitted* content which gave $r_s \sim 0$. These types of content reduced the performance of machine extraction algorithms.

Thus, as a general rule, concept map mining from lecture slides provided promising machine performance for *well-fitted* course content and hence, confirmed the main hypothesis of this research ‘*It is possible to develop auto-generated concept maps of lecture slides to strongly correlate with human constructed maps*’.

This study highlighted the importance of structural features in contrast to natural layout or the grammatical structure of text to determine the importance of concepts. The structural features considered in this study were *degree centrality*, *term frequency*, *co-occurrence*, *proximity* and *typographical information*. Therefore, it is important to consider how each of these features contributes to produce a high-quality concept map from lecture slides. For instance, when important domain concepts are emphasised (*typographical information*) and recapped (*term frequency* and *co-occurrence*), those concepts will be extracted using CMMF. In addition, a construction of probable links with the central idea of the topic and between other concepts (*degree centrality* and *proximity*) will improve the meaningful learning as well as the machine extraction.

The following feedback was received from lecturers during the study.

“I tend to think that summary generally contains things that have already been discussed. But, I found a new concept in the summary which hasn’t seen in the lecture slides. I read the lecture from the beginning again to locate that concept, but couldn’t find it”.

This comment provided an indication that there can be disjoint concepts included in lecture slides which are not fitting with students’ knowledge structures.

“There are tables which provide comparison between important concepts. How is this handled by the system?”

This is one of the limitations of this research. The data in tabular form might include useful domain concepts. However, a feature to tackle the comparisons in tabular data has not yet implemented.

“Examples are very useful to learn concepts, but they are not concepts. Therefore, I am not sure whether they should be included or not. I have included them in cases where I think they are very useful”.

“In IP, many domain concepts are introduced via analogy. So, are they also to be classified?”

Examples or analogies can be included into the extracted concept map, if they are strongly correlated with the domain or emphasised in the context.

When ranking the triples according to their importance in order to filter out trivial triples for knowledge organisation, CMMF only considered ‘concepts’ and not ‘relations’. It was assumed that ‘*if the participating concepts are selected as ‘important’, this implies that the connections between them are deemed important*’. When considering participating concepts, there can be situations where one of the concepts in the triple is important and not the other one. In such occasions, if the ‘starting node’ (i.e. subject) is the one which selected as important, the triple is considered to be ‘important’. In contrast, if the ‘ending node’ (i.e. object) is selected as ‘important’, such triples were placed in a special list called ‘*threshold triples*’. When calculating the threshold for filtering out irrelevant triples, if an average number of concepts per concept map could be increased, the triples in the ‘threshold triples’ list might consider according to the ranking order.

6.1.8 Summary

A series of studies were conducted to evaluate the effectiveness of CMMF. The three main studies focused on evaluating elements of concept map such as concepts and triples and the concept map as a whole. Another two supplementary studies were conducted to evaluate the algorithms for pronoun resolution and noise detection to support the performance of CMMF.

The initial evaluation study was conducted to test the hypothesis that concepts extracted from computer-based algorithms can be used as an alternative to human extracted domain concepts. The results of this study were not promising (F-measure was 43%) due to some limitations discussed in Section 4.2.4. Based on the performance improvement factors suggested, a second study was conducted to measure the effectiveness of triple extraction algorithm.

This study found that for some courses a F-measure greater than or equal to inter-rater agreement was achieved, supporting the hypothesis, while some courses had F-measure less than inter-rater agreement. However, an improved F-measure of 69% for triple extraction was achieved compared to 43% in concept extraction.

The final study evaluated the ranking of concepts and concluded that the ‘location’ of concepts within the layout of lecture slides (i.e. *title, bullet-point*) or the ‘grammatical structure’ of the concepts (i.e. *noun phrase, simple sentence*) had no influence in determining which concepts were important to learners. However, structural features and distributional features (e.g. *term frequency, number of degree centrality, proximity*) had a positive effect on determining which concepts were more important to the learner. A strong positive correlation ($r_s > 0.7$) was

achieved for some Computer Science courses. Therefore, this study concluded that the courses classified as *Software Engineering, Computer Architecture, Communications and Security* (ACM, 2012) which had rich grammar, complete sentences with apparent independent clauses, infrequent use of complex sentences, or confusing idioms were *well-fitted* for concept map extraction from lecture slides using CMMF.

The remainder of the chapter utilises the results obtained from the algorithm evaluation to select a Computer Science course for further evaluation of CMMF with students.

6.2 Evaluation of the Task-Adapted Scaffolding Framework

The previous section of this chapter discussed the quality of auto-generated concept maps by comparing with lecturers. The results concluded that CMMF is applicable for any course within the Computer Science domain, but is, best suited for courses classified as *well-fitted* contents.

The objective of this section of the thesis was to apply the CMMF into a real world application, particularly within the educational context and confirm whether the research demonstrates its intended behaviour. The purpose of this experimental study was to investigate the effect of different forms of concept maps on students' learning outcome using the research question;

'Does task-adapted concept map as scaffolding improve learning gain over other forms of scaffolding?'

6.2.1 Experimental Design

A randomised experimental design was used to answer the primary research question (Figure 6.3). The independent variable in this study was the 'type of scaffolding resource' such as lecture slides or the form of concept map. The dependent variable was the 'students' learning gain', which was calculated using pre and post-tests. In addition to the primary research question, a correlation analysis was conducted to measure whether there was any significant correlation between the time spent on scaffolding and the improvement. A mixed method of data collection was used, combining quantitative and qualitative techniques. Quantitative methods collected data from the system interactions, logs, answers to the questions in main study and post-test and closed questionnaires. Qualitative data was obtained from an open-ended questionnaire and observations during the experiments.

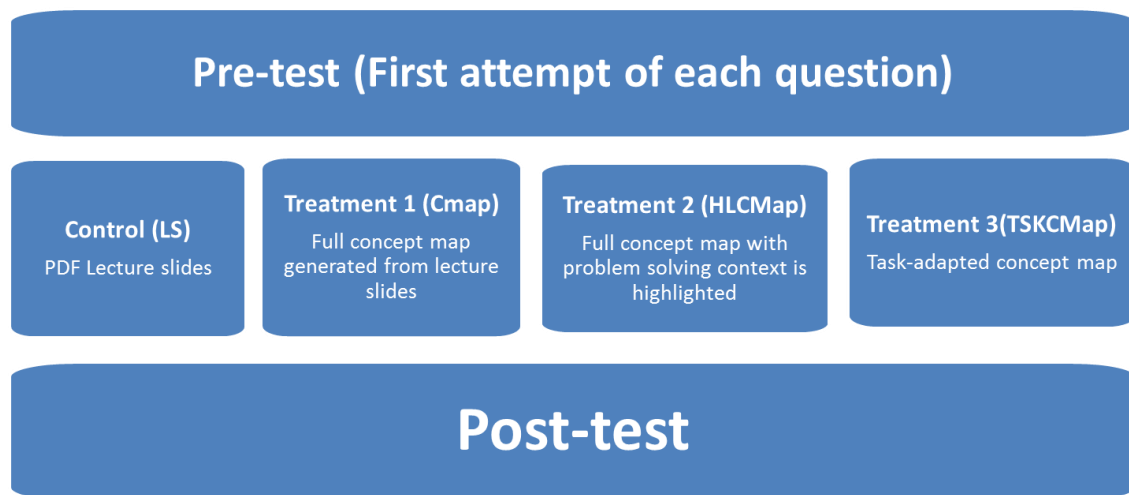


Figure 6.3: Experimental design

Subject Domain

The algorithm evaluation of the CMMF (Section 6.1) presented the results of lists of CS courses which had strong positive correlation ($r_s \sim 1$) with expert maps. From that list, the *Introduction to Software Engineering* course (*COMP SCI 2006*) was selected for the experimental study since it demonstrated strong positive correlation with expert maps ($r_s = 0.83$). This course also had tutorial sessions in computer labs which made it possible to arrange the computer-based study. This arrangement helped students to attend the study during the already scheduled tutorials.

The Software Engineering course contains a large set of new concepts that students are required to assimilate in their second year of undergraduate studies. These concepts are then applied to develop software products during following undergraduate years or in industry. Therefore, Software Engineering is a core Computer Science course. For the experiments, the ‘Software testing’ topic was selected which includes a large set of concepts related to testing a software product.

Participants

The participants in this study were Undergraduates of the University of Adelaide aged over 18 years, who had enrolled in the *Software Engineering* course in semester 1, 2014. The majority of the students were majoring in Computer Science, while others were from various Engineering disciplines. The experiment did not collect gender or other demographic details of the students. However, the majority of the class were male students.

Participant recruitment was first announced by the lecturer in week 3 and week 5 for the main study and the post-test respectively. After the class announcements, a notice was published on

the Learning Management System. The recruitment notice basically included eligibility criteria, a short description of the research and the contacts, format of the study, date, time and venue, expected completion time and the voluntary nature of the study. A reminder was posted on the day before the experiments. 62 out of 105 enrolled students (59%) participated in the study.

Materials

The materials were categorised into three groups: pre-experimental materials, experimental materials and post-experimental materials.

Pre-experimental materials

The pre-experimental materials are included in Appendix C. Student were given coloured set of papers (*'yellow' for control, 'blue' for treatment 1, 'green' for treatment 2 and 'pink' for treatment 3*) based on their experimental groups. This set of papers included 'instruction sheet', 'consent form', 'concept map example', 'information sheet', 'independent complaint form' and the 'questionnaire'. The instruction sheet contained simple steps to follow during the study such as requesting them to read and sign the consent form as the first thing, and how to use the web-based system (see Appendix C).

There was no formal concept map training provided to the treatment groups. Instead, they were presented with a paper-based concept map example (Appendix C). Additionally, five HDR students were made available during the study to assist participants with concept map related issues. The concept map example included a simple definition of a concept map and its elements, such as concepts, relations and relation labels. A drawn concept map about 'cats' was used to illustrate the idea (Appendix C).

The Information sheet included the information about the research project, aims, nature of the study, risks and whom to be contacted regarding the research. The Independent complaint form included contact details of the Human Research Committee. This research was approved by the Human Research Ethics Committee of University of Adelaide (**Project approval no: H-2014-052**).

Experimental materials

The experimental materials included quizzes, and the resources used for scaffolding. All the groups received the same set of quizzes; however, they received different forms of scaffolding.

Quiz

Ten multiple choice questions (MCQ) were constructed for the experimental study. These questions and answers covered 68% of the important domain concepts in the 'Software testing'

topic. This percentage was calculated by comparing the concepts ranked by the lecturers in a previous study (Section 6.1.7). The construction of questions followed the approach discussed in Guru ITS (Person et al., 2012). According to Guru ITS, the concept maps from the topic were generated and afterwards, identified concepts and relations to construct questions. In order to determine the question type, this research utilised objective levels of Bloom's taxonomy (Bloom et al., 1956). Therefore, 'key words' of Bloom's taxonomy were used as part of the question stems (See Table 2.3) (Appendix C). Finally, the questions were modified by correcting grammar and rephrased to make them more meaningful to the students. Additionally, text book quizzes and previous examination questions were utilised to improve the set of MCQs. The constructed questions and answers were reviewed by the course coordinator of 'Software Engineering' and a domain expert.

Scaffolding resources

According to the experimental design in Figure 6.3, each group received a different form of scaffolding resource. The control group received lecture slides (LS) as a PDF file. This file had fewer slides than the original slide set. A subset of the original slide set which was related to each question was extracted manually. This process reduced the disadvantage faced by the control group having more content than the treatments.

Treatment group 1 (CMap) and 2 (HLCMap) received the concept maps generated from lecture slides using the CMMF discussed in the Chapter 4. Treatment group 1 (CMap) received the full concept map of 'Software testing' topic. Treatment group 2 (HLCMap) received the same concept maps as group 1; however, in their maps, the context to answer the question was highlighted. The task-adapted concept map of the same question was utilised to find the corresponding context in HLCMaps. The highlighting process included applying a different background colour and a shadow effect and increasing the font size of the corresponding set of concepts (Appendix C). This process was performed manually using the IHMC CMap tools (Canas, Hill, et al., 2004). The aim of introducing the HLCMap group was to measure whether there was any effect of having scaffolding in between full concept maps and task-adapted concept maps.

Treatment group 3 (TSKCMMap) received task-adapted concept maps extracted from *TASF*, as discussed in Chapter 5. All the scaffolding resources were pre-processed for the experiments and stored as image files or as PDF files. A sample of resources used for the experiments is included in Appendix C.

Post-experimental materials

Post-test

The preparation of post-test questions followed the same process as MCQs for the main study. However, post-test questions included combinations of MCQs, fill-in-the-blanks and open-ended questions to minimise the opportunity to guess, allowing for the comparison of actual learning gain between pre- and post-tests. The post-test was arranged as paper-based tests. In the post-test, there were three similar questions as main study, but they were rephrased and shuffled to reduce the possibility of memorisation. The remaining questions were new, but from the same ‘Software testing’ topic (Appendix C).

Questionnaire

The questionnaire contained 10 questions with a combination of 5-point Likert scale (ranging from ‘strongly disagree’ to ‘strongly agree’) and open-ended questions (Appendix C). Students in the control group who have not seen concept maps received a slightly different version of the questionnaire. The questionnaire was organised to collect data on following areas;

- Prior experience on knowledge organisations
- Feedback on concept mapping as a learning technique
- Feedback on CMMF
- Feedback on TASF
- Courses that students would like to integrate concept maps in future studies and other suggestions for improvements

Procedure

Prior to the experiments, a pilot process was carried out to test the web-based prototype. A group of HDR colleagues were invited to test the prototype and report any issues. The colleagues tested the system on different web browsers such as *Firefox*, *Internet Explorer*, *Google Chrome* and *Safari*. Preliminary testing indicated that the best browser for the experiments was Firefox with automatic PDF download disabled.

The prototype contained a link in the ‘scaffolding’ web page to ‘go back to the question’ (see Figure 5.11 (c)). The time spent on the scaffolding until participants clicked the ‘go back to question’ link was recorded. However, some students in the pilot experiments clicked the ‘back’ button in the web browser instead of ‘go back to question’ link. Therefore, the same functionalities were implemented for both the ‘back’ button in the browser and ‘go back to

question' link. After fixing these issues, the prototype was tested on 40 machines in a computer suite.

During the main study, participants were required to sign the written consent form, giving their permission to collect their data, store and analyse and anonymously use their data for this thesis and any publications written from this work. Students were also notified that participation was voluntary and they could exit the study at any time, at which point, their data would be completely removed from the analysis.

Upon log in to the system, participants were given a maximum of 20 minutes to attempt the 10 MCQ quizzes. As explained in the Section 5.4, if the attempted answer was correct, students were directed to the next question. Otherwise, students were given an option to get 'help' from the system. The help was specific to the experimental group to which they were assigned (see Figure 6.3). Once the participants completed learning through scaffolding, they were required to go back to the initial question for which they requested help. However, the answers were shuffled at this stage to reduce guessing. This process could be repeated any number of times, but after three unsuccessful attempts in a question, students were given an option to skip the question to reduce frustration.

After the quiz task, students were requested to complete the questionnaire indicating the issues, if any, they encountered, their opinions about the system, new features they wanted to see and their willingness to use the system for future studies.

During the study, five HDR students were engaged to collect observations of the participants. The participants were sent to the lab on arrival basis by assigning a number (ArrivalID) starting from 1. Five observers recorded the seatings of the participants based on their ArrivalID. Then, their behaviours were recorded using visual annotations against ArrivalID. A sample of annotations used by the observers is shown in Table 6.13.

Table 6.13: Visual annotations used by observers

Very keen	Look away from task	Nervous or confuse	Bored	Scrolling for no reason	Go back and forth	left early	Delayed start
KEN	AWY	NER	BRD	SCR	BCK	LFT	DEL

After the experiments, the student IDs were replaced with pseudonyms and the same pseudonyms were assigned to ArrivalIDs. This information was utilised only when anomalies were discovered in data. For instance, some participants had not completed all the questions.

Then, their behaviours were mapped using the observation sheet to understand whether they started the study behind the schedule or left early.

Participants who were involved in the first stage of the experiments were invited to participate in the post-tests. Post-tests were carried out after two weeks from the main study. Post-test were expected to complete within 20 minutes (2 minutes per question). Help was not provided to any of the groups during the post-test.

6.2.2 Data Analysis

After the main study, all the data was downloaded from the MySQL database and student IDs were replaced using pseudonyms. The results of students' first attempt to each MCQ quiz were extracted as the pre-test score. If the first attempt was correct, 1 point was assigned and 0 otherwise.

The post-test was assessed manually since it was paper-based. One point was assigned for correct answers and 0.5 for partially correct answers in *open-ended* questions and 0 otherwise. The same pseudonyms were used to replace student IDs in post-tests.

Apart from the quantitative data, answers to questionnaire in terms of 5-level Likert scale and open-ended questions were collected for qualitative data analysis. The questionnaire answers were completely anonymous.

SPSS statistical software (IBM SPSS 22) was used for data analysis. This research investigated following research questions.

- 1) *Does task-adapted concept map as scaffolding improve learning outcome over lecture slides as scaffolding?*
- 2) *Does task-adapted concept map as scaffolding improve learning outcome over full concept map as scaffolding?*
- 3) *Does task-adapted concept map as scaffolding improve learning outcome over full concept map with highlighted problem solving context as scaffolding?*
- 4) *Is there any significant correlation between the time spent on scaffolding and the post-test scores?*
- 5) *What are the students' opinions of the research framework?*

The first three questions were analysed using Analysis of Variance (one-way ANOVA) by comparing the scores between pre and post-test. The alpha level was set to 0.5 ($p \leq .05$).

Research question 4 was analysed using Pearson correlation and a qualitative analysis was conducted to answer final research question.

6.2.3 Results and Discussion

Prior to discussing the results of the experiments, factors that might influence the results were considered. Initially, students were assigned to groups randomly in an even distribution. However, some data was precluded from analysis since some students were behind schedule and did not complete all the questions. Actual number of participants considered in the data analysis is shown in Table 6.14.

Table 6.14: Statistics of student groups

Group	Description about instructional scaffold	Number of students
Control (LS)	Lecture slides as PDF	11
Treatment 1 (CMap)	Full concept maps extracted from lecture slides	15
Treatment 2 (HLCMap)	Full concept map (problem solving context highlighted)	16
Treatment 3 (TSKCMaP)	Task-adapted concept maps	17
Total		59

The prior experience of students' on knowledge organisation techniques was measured from the collected questionnaire data.

Table 6.15: Students' prior experience on knowledge organisations techniques

Prior experience	Technique	Number of students
No prior experience		10
Have heard, but not used		11
Previously used	Concept maps	11
	Mind maps (e.g. inspiration)	21
	Semantic networks	1
	Simple brainstorming for ideas	1
Currently using	Concept maps	3
	Mind maps (e.g. popplet.com)	10
	Knowledge maps	1

From the statistics, 18% ($n = 10$) had no prior knowledge or experience on knowledge organisation techniques. Another 20% ($n = 11$) had heard about concept maps, mind maps or similar techniques. The remaining 62% participants ($n = 35$) had previously used or are currently using, knowledge organisation techniques such as mind maps and concept maps (see Table 6.15).

Pre-test

The main study was conducted in week 4 after the relevant lecture topic was completed in week 2 with a two week gap. The quizzes covered in the experiments were not discussed purposely in any tutorials or as review questions during the lecture. According to the *forgetting curve* proposed by Ebbinghaus (1913) and the recent works of Schacter et al. (2009), students generally forget what they have learned in the lecture within 6-7 days unless revised frequently. Therefore, after two weeks from the lecture, it was assumed that there was no gap of the knowledge between students who attended the lecture and others unless the former group revised the contents regularly.

The response to the first attempt of each question was considered as students' prior knowledge on the 'Software testing' topic. Table 6.16 illustrates means, medians and standard deviations of pre-test scores per group. The total score of each student ranged between 0 and 10.

Table 6.16: Descriptive statistics of pre-test scores

Group	Mean (M)	Median (Md)	Standard deviation (SD)	n
LS	2.9	3	1.9	11
CMap	1.8	2	1.5	15
HLCMap	2.6	3	1.5	16
TSKCMMap	2.5	3	1.5	17
Total	2.4	2.7	1.6	59

Pre-test scores are further illustrated using a histogram in Figure 6.4. The statistics indicated that the majority of the participants did not have the relevant prior knowledge ($M = 2.4$, $SD = 1.6$, $n = 59$).

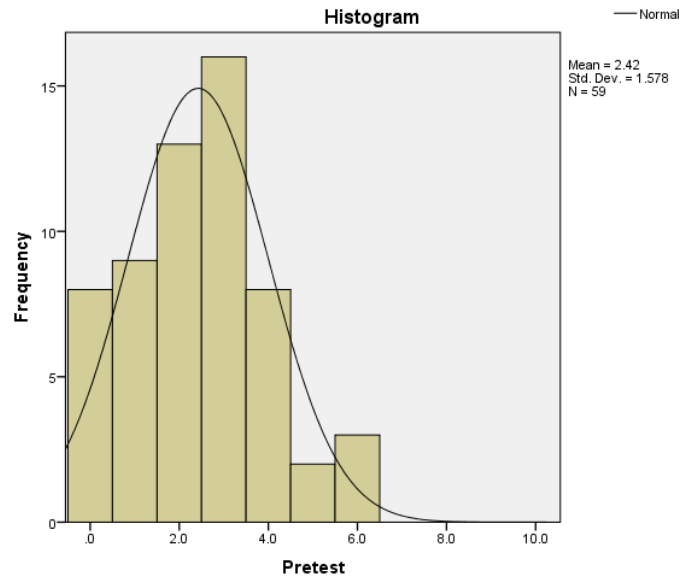


Figure 6.4: Histogram of pre-test scores

Main Study and Post-test

Data collected from the sample of 30 students who participated for the post-tests were used to calculate the 'learning gain' between post- and pre-test scores. The descriptive statistics of students' learning gain across different scaffolding groups are shown in Table 6.17.

Table 6.17: Descriptive statistics of learning gain

Group	Mean (M)	Standard deviation (SD)	Standard error	n
LS	2.5	2.0	.755	7
CMap	3.2	1.4	.532	7
HLCMap	3.6	2.0	.777	7
TSKCMaP	5.0	1.5	.496	9
Total	4.0	1.9	.351	30

The results illustrate that the TSKCMaP group had the numerically highest mean ($M = 5.0$, $SD = 1.5$, $n = 9$) while the control group (LS) had the smallest mean ($M = 2.5$, $SD = 2.0$, $n = 7$). The other two groups had their mean scores in-between TSKCMaP and LS groups.

One-way ANOVA was conducted to compare means within and between groups. There are three assumptions required to be satisfied prior to conducting ANOVA.

First assumption: *homogeneity of variance between groups*

The most common approach to test this assumption is to conduct Levene's test. If the resulting *p-value* of Levene's test is greater than the significance level determined by the statistical tests ($p = .05$), then the Levene Statistic is not significant. This means that there are comparable variances among the different groups and the homogeneity requirement is satisfied. Table 6.18 illustrates the results of Levene test.

Table 6.18: Test of homogeneity of variances (learning gain)

Levene Statistic	df1	df2	Sig.
.698	3	26	.562

According to the results, test of homogeneity assumption was not violated; $F(3, 26) = .698, p = .562 (> .05)$.

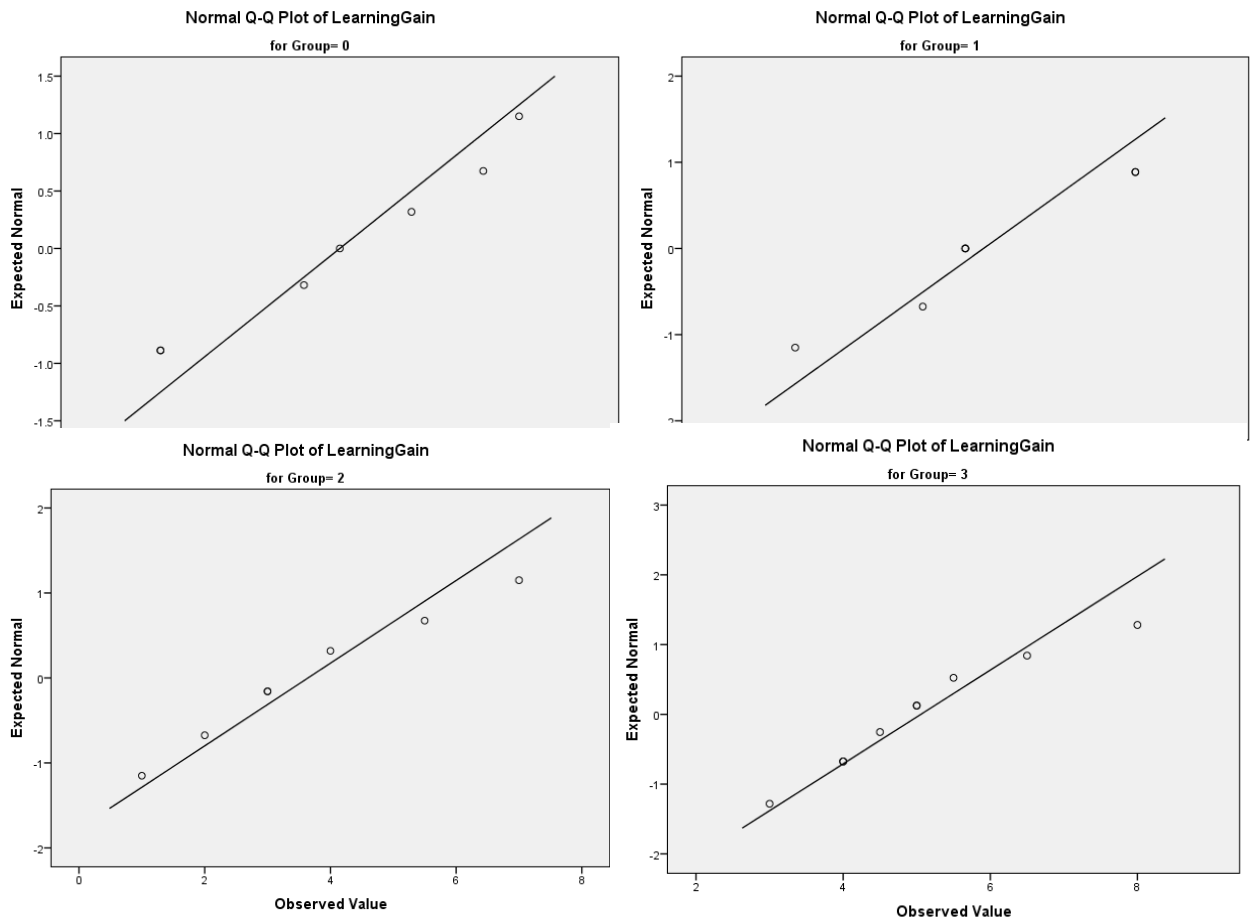
Second assumption: *dependent variable is normally distributed around means*

In order to test this assumption, tests of normality have been conducted (See Table 6.19).

Table 6.19: Tests of normality

Group		Shapiro-Wilk		
		Statistic	df	Sig.
Learning gain	0	.917	7	.445
	1	.885	7	.252
	2	.963	7	.840
	3	.948	9	.668

According to the results, the dependent variable (learning gain) is normally distributed across different scaffolding groups ($p > .05$). An output of a normal Q-Q plot demonstrates the normality graphically in Figure 6.5. The data points close to diagonal line and the Sig. value of Shapiro-Wilk test indicates that the normality assumption was met.



Third assumption - independence of observation

This assumption suggests that an individual in one group’s performance is not influenced by others in the same group. There is no influence of one individual performance to other in the same group since individuals were assigned to groups in a random order. Thus, third assumption was also not violated.

Since the three assumptions of ANOVA were not violated, one-way ANOVA was conducted to compare means within and between groups (Table 6.20).

Table 6.20: Summary results of ANOVA

	Sum of Squares	df	Mean square	F	Sig.
Between Groups	28.292	3	9.431	3.103	.044
Within Groups	79.008	26	3.039		
Total	107.300	29			

Results of the one-way ANOVA indicated that the means between groups were significant; $F(3, 26) = 3.103, p = .044 (< .05), \eta^2 = .263$. However, the summary table (Table 6.20) does not demonstrate which pairwise comparisons are significant. Post-hoc test was conducted to compare the means of difference groups. This study utilised Tukey HSD post-hoc test since it allows for multiple pairwise comparisons without an increase in the probability of a Type 1 error.

Hypotheses Testing

Results of Tukey HSD post-hoc test are presented in Table 6.21.

Table 6.21: Summary of Tukey HSD post-hoc test

(I) Group	(J) Group	Mean Difference (I-J)	Std. error	Sig.	95% Confidence Interval	
					Lower Bound	Upper Bound
LS	CMap	-.71429	.93178	.450	-2.6296	1.2020
	HLCMap	-1.14286	.93178	.231	-3.0582	.7724
	TSKCMMap	-2.55556*	.87849	.007	-4.3613	-.7498
CMap	HLCMap	-.42857	.93178	.649	-2.3439	1.4867
	TSKCMMap	-1.84127*	.87849	.046	-3.6470	-.0355
HLCMap	TSKCMMap	-1.41270	.87849	.120	-3.2185	.3931

*. The mean difference is significant at the 0.05 level

The testing results of each hypothesis are reported below followed by a discussion of the results.

Hypothesis 1

Students who receive task-adapted concept maps (TSKCMMap) as scaffolding will have a higher learning gain than the students who receive lecture slides (LS) as scaffolding.

To test this hypothesis, mean scores of learning gain from the TSKCMMap group and the LS group were compared. The results showed that the TSKCMMap group obtained higher learning gain ($M = 5.0, SD = 1.5, n = 9$) than LS group ($M = 2.5, SD = 2.0, n = 7$). The results were statistically significant based on the results of post-hoc test; $p = .007 (< .05)$. Therefore, results of the one-way ANOVA supported the hypothesis that students who received task-adapted concept maps (TSKCMMap) as scaffolding would perform better than students who received lecture slides (LS) as scaffolding. However, due to the smaller sample size in each group ($n = 9$ in TSKCMMap and $n = 7$ in LS), these findings cannot be generalised to a wider population (Kenny, 1987).

The results also demonstrated that the learning gain using Concept Maps (CMap) ($M = 3.2$, $SD = 1.4$, $n = 7$) as scaffolding was numerically higher than the LS group ($M = 2.5$, $SD = 2.0$, $n = 7$). However, this difference was not significant; $p = .45$ ($> .05$). Concept maps as a learning technique have been compared with text representations in various studies. Hall & O'Donnell (1996) and O'Donnell et al. (2002) found that utilising knowledge organisation techniques such as knowledge maps or concept maps, in contrast to text representations, possessed significantly higher performance. A research with 43 participants in the form of knowledge maps (i.e. treatment group, $n = 22$) or text representations (i.e. control group, $n = 21$) studied 1500 words on autonomic nervous system and rated their motivation, anxiety and concentration using 10 rating scale. Afterwards, students completed a performance test. The results demonstrated significantly higher scores for recall ideas, subjective concentration and motivation in the knowledge map group compared to the traditional text group (Hall & O'Donnell, 1996).

Another research on knowledge mapping compared high and low ability students who were taught using knowledge maps or traditional text. The results showed that low ability students taught using knowledge maps performed better than the control group. However, high verbal ability students did not vary based on the 'medium' of the study. In addition, students recall more central ideas when they learned using knowledge maps (Patterson et al., 1993). The choice of utilising knowledge maps or traditional text also depends on learner's prior knowledge. Students with low prior knowledge on Biology learned most when lecture was accompanied with knowledge maps. Higher prior knowledge students had no effect on the medium (Lambiotte & Dansereau, 1992).

Hypothesis 2

Students who receive task-adapted concept maps (TSKCMaP) as scaffolding will have a higher learning gain than the students who receive full concept maps (CMap) as scaffolding.

To test this hypothesis, mean scores of learning gain from the TSKCMaP group and the CMap group were compared. The results showed that the TSKCMaP group obtained significantly higher learning gain ($M = 5.0$, $SD = 1.5$, $n = 9$) than the CMap group ($M = 3.2$, $SD = 1.4$, $n = 7$); $p = .046$ ($< .05$). Therefore, results of the one-way ANOVA supported the hypothesis that students who received task-adapted concept maps (TSKCMaP) as scaffolding performed better than students who received full concept maps (CMap) as scaffolding.

The primary reason for this could be the amount of information included in each concept maps and their relevancy to the problem solving context. Task-adapted concept maps contained the information that is most relevant to solve the problem while full concept map represented the domain. According to Canas & Novak (2006), concept maps construct to answer a question

(called the 'focus' question) is more effective than concept maps constructed in represent a domain. The former process involved more dynamic thinking and a deeper understanding than the latter process.

This finding was supported by an empirical study conducted by Eylon & Reif (1984). Their study suggested that higher levels of the hierarchy should preferentially contain information most important for the domain of tasks. Their study compared the effectiveness of two hierarchical knowledge organisations that contained the same knowledge, with one of them adapted to a set of given tasks. The results demonstrated that the treatment group who received task-adapted information performed significantly better than the control group who received hierarchical information without task adaptation.

Hypothesis 3

Students who receive task-adapted concept maps (TSKCMaP) as scaffolding will have a higher learning gain than the students who receive full concept maps with highlighted problem solving context (HLCMaP) as scaffolding.

To test this hypothesis, mean scores of learning gain from the TSKCMaP group and the HLCMaP group were compared. The results showed that the TSKCMaP group obtained higher learning gain ($M = 5.0$, $SD = 1.5$, $n = 9$) than the HLCMaP group ($M = 3.6$, $SD = 2.0$, $n = 7$). However, the results were not statistically significant; $p = .120$ ($> .05$). Therefore, results of the one-way ANOVA did not support the hypothesis that students who received task-adapted concept maps (TSKCMaP) as scaffolding performed better than students who received full concept maps with highlighted problem solving context (CMaP) as scaffolding.

This could be due to the fact that even though the HLCMaP group received full concept maps as similar to the CMaP group, the participants in the HLCMaP group might only have looked at the highlighted area of the map without being overloaded by the number of concepts and relations provided in the full concept map. The idea of highlighting more relevant information to the context was well supported from the feedback of students in the CMaP group. These students repeatedly mentioned the importance of having different colours or switches to differentiate more relevant information within the concept map. This will be discussed under the 'qualitative analysis' section.

Although several studies in related literature provided elements of concept maps (e.g. concepts, relation labels) as scaffolding for learners, the focus of these studies was to assist constructing concept maps manually but not facilitate problem solving by providing full or task-adapted concept maps as scaffolding (Canas, Carvalho, et al., 2004; Chang et al., 2001; Leake et al.,

2003; Ruiz-Primo et al., 2001; Weinbrenner, Engler, & Hoppe, 2011). Even though manual construction of concept maps has a high learning gain ($d = .82$) than students studying concept maps ($d = .37$) (Nesbit & Adesope, 2006), constructing concept maps as a learning technique is not widely popular in tertiary education in comparison to K-12 education (Bagno & Eylon, 1997; Leelawong & Biswas, 2008; McClure et al., 1999; Okebukola, 1992; Pankratius, 1990; Person et al., 2012) due to the problems associated with manual construction of concept maps as discussed in the previous chapters including additional workload and high intellectual commitment. Thus, facilitating problem solving through scaffolding via task-adapted concept map is indicated to be a well-suited learning technique within *tertiary education*.

Correlation Study

The objective of this study was to investigate the research question;

‘Is there any significant correlation between the time spent on scaffolding and the learning gain?’

Pearson correlation analysis was conducted to answer this research question and the results are reported in the Table 6.25. The data was normalised by eliminating the amount of time spent is less than 5 seconds since these data basically included students clicking back and forth between the ‘help’ and the quizzes.

Table 6.22: Pearson correlation between the time spent on scaffolding and learning gain

Group	Correlation	Sig. (2-tailed)	N
LS	-.413	.309	9
CMap	.594	.042	12
HLCMap	.102	.810	13
TSKCMaP	.801	.003	13

There was a positive correlation between time spent on scaffolding and learning gain, in two of the treatment groups, TSKCMaP ($\gamma = .801$, $p < .05$) and CMap ($\gamma = .594$, $p < .05$). This finding was supported by the work of Person et al. (2012) which suggested that the time spent on scaffolding and learning gain was positively correlated. However, the LS group had a negative correlation between the time spent and learning gain. This could have occurred if the students in the control group spent more time scrolling the lecture slides to formulate an answer when the relevant information was scattered throughout the slides. According to Martin (2008),

knowledge organisation of the contents of the educational materials help students find, compare and memorise the information scattered in these materials.

Qualitative Study

The objective of this study was to investigate the research question;

‘What are the students’ opinions of concept mapping and scaffolding?’

The main findings from the students’ responses to the questionnaire are discussed below. The statistics of the findings are included in Appendix D.

What do you think about concept maps/lecture slides used in this study to answer questions?

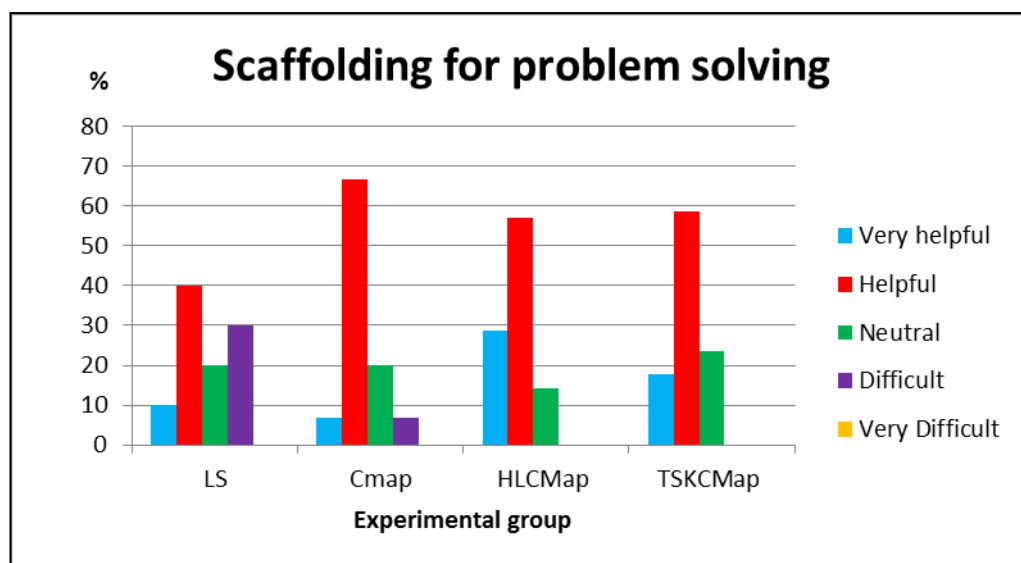


Figure 6.6: Students’ opinion on having concept maps or lecture slides as scaffolding to answer questions

Based on the statistics, the treatment groups were very positive about having concept maps as scaffolding for problem solving (see Figure 6.6). Among the participants of the treatment groups, 77%, 86% and 73% stated it was either ‘helpful’ or ‘very helpful’ to have concept maps as scaffolding in their groups. In the LS group ($n = 10$) 30% of students mentioned that it was difficult to utilise lecture slides to answer questions.

If it was difficult to use concept maps in this study, please select the statement (s) that describes your problem

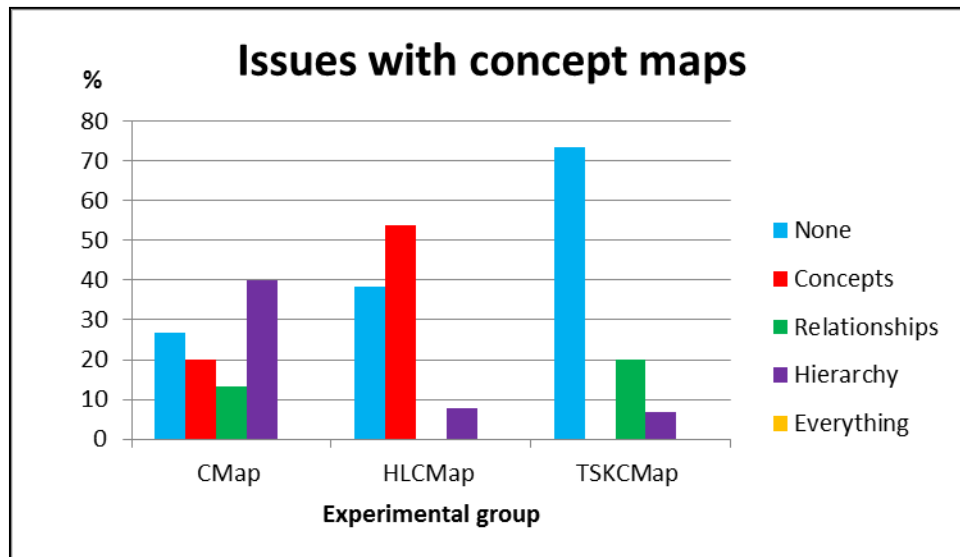


Figure 6.7: Students' issues related to concept maps

Based on the statistics, 73% of students from the TSKCMap group ($n = 15$) had no issues with concept maps, suggesting that the structure and the smaller number of concepts of task-adapted concept maps might prove to be simpler to these students (see Figure 6.7). However, 20% of students from the TSKCMap group had issues with identifying 'relationships'. This might be due to the fact that some disjoint concepts included in task-adapted concept maps might make it difficult for students to organise knowledge. In contrast, the HLCMap group ($n = 13$) had no issues with relationships while in the CMap group; only 13% had issues with relationships.

In this research, we developed a tool to automatically generate concept maps from lecture slides. "This kind of tool will help answering questions"

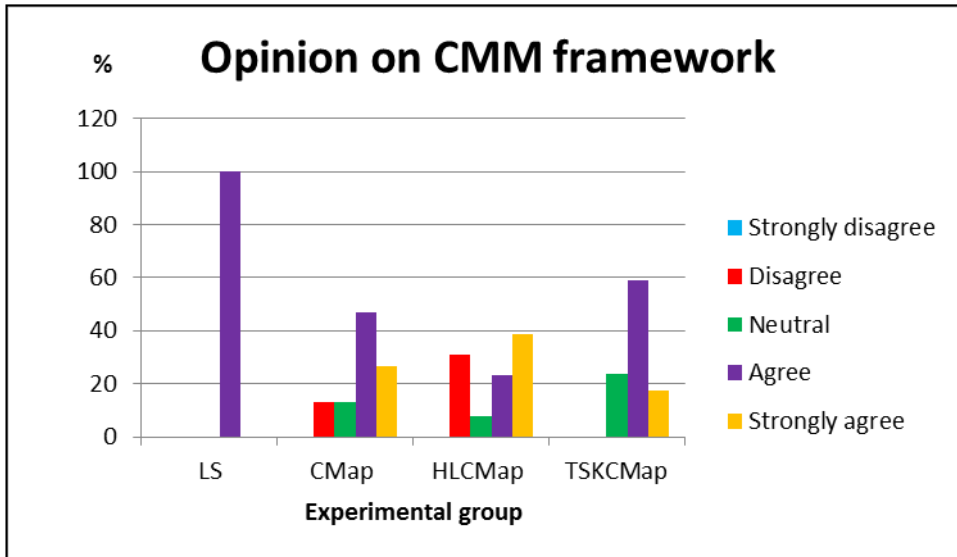


Figure 6.8: Students' opinion on CMMF

According to the statistics, students in the control (LS) group ($n = 10$) all agreed to have automatically generated concept maps from lecture slides for answering questions (see Figure 6.8). Other three treatment groups also have positive response including 73%, 62% and 76% of participants either agreed or strongly agreed in CMap ($n = 15$), HLCMap ($n = 13$) and TSKCMap ($n = 17$) groups respectively.

Which form of resources do you prefer if our tool made available through CS Forums in future?

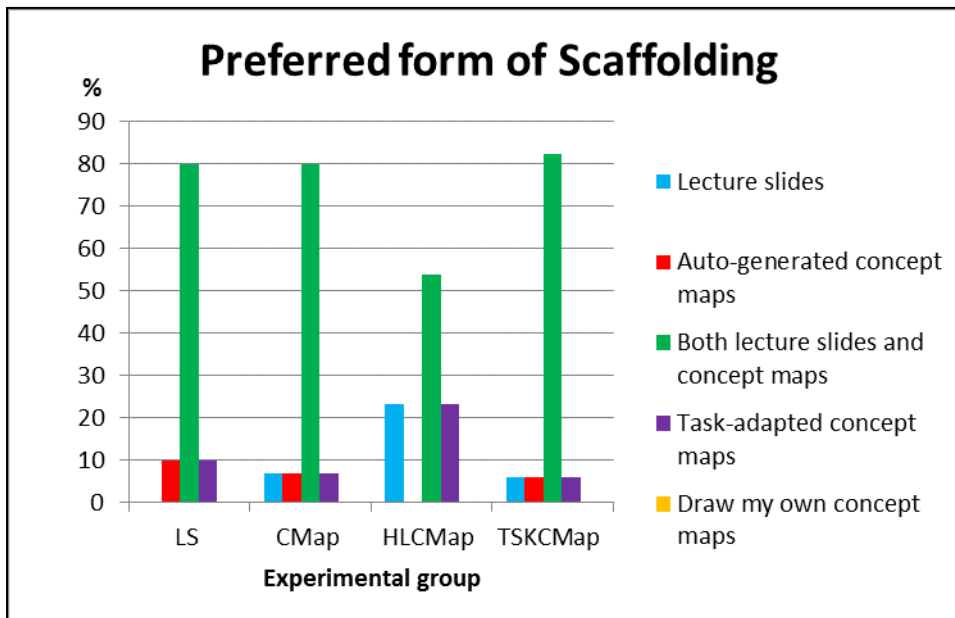


Figure 6.9: Students' preferred form of scaffolding

The statistics demonstrated that the students are extremely receptive to the use of concept maps and lecture slides for their studies including 80% in the LS group ($n = 10$), 80% in the CMap group ($n = 15$), 54% in the HLCMap group ($n = 13$) and 82% in the TSKCMap group ($n = 17$). It appears students believe utilising concept maps generated from lecture slides would be useful as a supplement to traditional learning approaches. The benefit of utilising knowledge organisation techniques as a supplement to lecture slides was supported by the studies of Brandt et al. (2001) and Kinchin et al. (2008). As stated in their work, it was challenging for lecturers to explicitly express complex knowledge structures created by them using point-based lecture slides. Therefore, it was difficult for lecturers to predict how students may interpret the information in the lecture slides and assimilate the content. It is likely that learners often construct false hierarchies which are not intended by lecturers (see Figure 2.3 and 2.4). Therefore, according to Kinchin (2006a) and Kinchin et al. (2008), a combination of lecture slides and concept maps contributes to an epistemologically balanced teaching approach.

In addition, none of the participants out of 55 were interested in constructing the concept maps manually. This feedback can be used to support the rationale of providing students with auto-generated concept maps as scaffolding for learning.

Would you like a tool which can extract partial concept maps to assist answering questions?

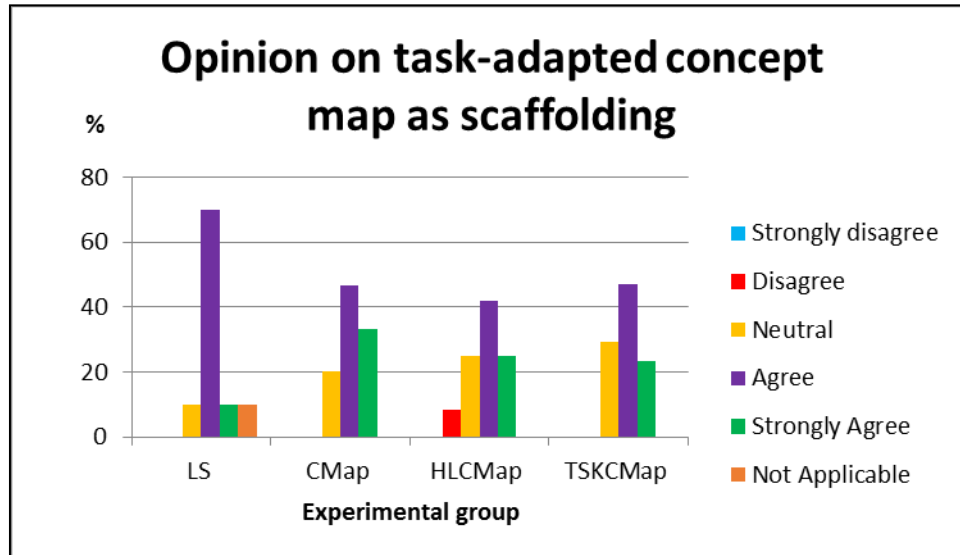


Figure 6.10: Students' opinion on task-adapted concept maps as scaffolding to answer questions

The results demonstrated that the majority of the students in each group either agreed or strongly agreed to incorporate task-adapted concept maps (80%, 80%, 67% and 71% in the LS ($n = 10$), CMap ($n = 15$), HLCMap ($n = 12$) and TSKCMap ($n = 17$) groups respectively) (see Figure 6.10).

What type of courses do you think this kind of tool will be more useful?

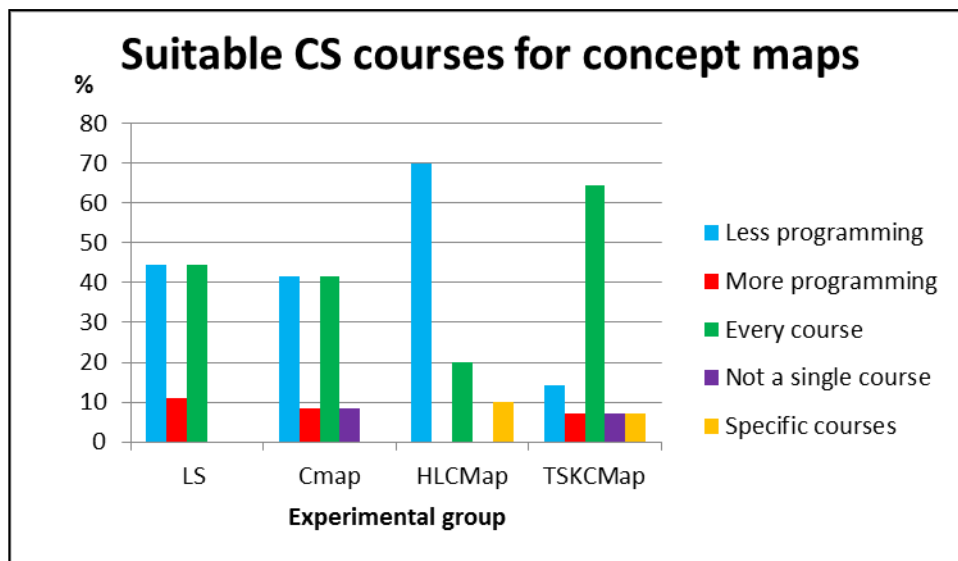


Figure 6.11: Most suitable Computer Science courses for concept mapping

According to the statistics, the majority of the participants in the LS group ($n = 9$) and the CMap group ($n = 12$) preferred to have concept maps for courses with less programming components or almost every course, including 89% in the LS group, 83% in the CMap group. In addition, 90% participants in the HLCMap group ($n = 10$) preferred to have concept maps for courses like Software Engineering and Operating Systems and 79% in the TSKCMap group ($n = 14$) chosen to have concept maps for every course (see Figure 6.11). Some participants mentioned that they would like to see concept maps for the courses with more facts or heavy theory or concepts, emphasising *‘based around memory information courses, courses with heavy theory/concepts which interconnects with each other’*. Based on this kind of feedback, it was clear that it would be beneficial for students to have concept maps generated from lecture slides to see the interconnections between concepts.

Finally, participants were questioned about the issues they had with the system and any suggestions for improving the system. Their feedback according to the groups is categorised in Table 6.26.

Table 6.23: Pros and cons about concept maps as scaffolding

Group	Feedback (Pros and cons)
CMap	<i>Too much information in concept map</i>
	<i>Some questions had no correct answer in the concept map</i>
	<i>Problems I had with the concept maps were that they were kind of bland. As more and more information is added to a concept map, it can be difficult to navigate</i>
HL CMap	<i>Larger maps are more difficult to read</i>
	<i>Some questions required more information than showing</i>
	<i>Concept maps were useful for hints</i>
TSK CMap	<i>Some maps did not easily show the information needed to answer the question</i>
	<i>Question 10's concept map was basically useless with help answering the question</i>
	<i>Not enough information provided in the concept tree</i>

According to the students' comments, it is correct that some concept maps did not have adequate information to answer all questions. The particular concepts and their relationships to learn the specific skills were included, but not the direct answer. In contrast, participants in the first two treatment groups (CMap and HL CMap) criticised about excess of information in full concept maps. This was one of the objectives of this thesis, not to give them direct answers, but provide the context to derive the answer. It is the responsibility of the students to meaningfully learn the fundamental concepts and their relationships to apply them to the problem solving context.

Table 6.24: Suggestions for improvements of the system

Group	Suggestions for improvement
CMap	<i>Removing timer or making optional, gives more time to think and less pressure</i>
	<i>Use colour for help identifying important sections</i>
	<i>Improve appearance by providing partial concept maps that applies to topic</i>
	<i>Perhaps be able to view the concept maps before answering the question</i>
	<i>Need colours and switches</i>
	<i>Less concepts</i>
	<i>The concept map can be improved more to have zoom in/out to smaller or larger sections</i>
	<i>Perhaps a way to toggle between maps of higher and lower densities of information</i>
	<i>Colour code hierarchies or smaller maps</i>
HL CMap	<i>In order to fix the problem of difficulty in navigating larger concept map, suggest to use colour to differentiate different components and branches</i>
	<i>Ability to search within the concept map, this would allow it to be useful for specific information or question rather than the full overview</i>
	<i>More easier to search</i>
	<i>A feature in which you can click on a concept to retrieve more information</i>
	<i>Add extra notes</i>
TSK CMap	<i>More details in relation labels</i>
	<i>Allow viewing the concept map at any time, not just when answering the question</i>
	<i>More explanations in concept maps</i>
	<i>It was quite good</i>
	<i>Being able to select on elements and have connected or related information and their paths highlighted</i>
	<i>Click on a specific concept or relations could bring up more details on that specific section</i>

From the feedback collected from the CMap and HLCMap groups, it is evident that there should be a mechanism to differentiate concepts that are more relevant to the context. Since, these two groups had no idea about task-adapted concept maps, they repeatedly mentioned the requirement of colour codes, 'search' option, zoom in/out or *smaller maps* that focus on the more relevant information to learning (see highlighted comments in Table 6.27). Although concept maps have been used as an 'intelligent suggester' to formulate queries to search in the Web space (Leake et al., 2004), research to search within a concept map to identify relevant information or colour-coding is lacking in the related literature. A preliminary study has been conducted to use colour codes to differentiate main ideas and identify similar ideas in the concept maps used in handheld devices. A questionnaire reported that 65% of the students ranked colour codes as the most helpful feature (Luchini et al., 2002). According to Table 6.27, all the groups were interested to have more details on specific concepts or relations by clicking them. A study to suggest multimedia resources, concept maps and propositions from the Web that are related to the elements of a concept map was proposed by Leake et al. (2003).

6.2.4 Summary

This section presented the evaluation of the task-adapted scaffolding framework with University undergraduates. Students were randomly assigned to three experimental groups and a control group. Each of these groups received a different form of scaffolding resources for problem solving. The control group received lecture slides as a PDF file. Treatment group 1 received full concept maps generated from lecture slides using the CMMF. Treatment group 2 received full concept maps with the context to answer the question highlighted. Treatment group 3 received task-adapted concept maps extracted using *TASF* which contained the related information to answer the given questions. Pre- and post-tests were compared between groups to obtain the learning gain. The results demonstrated that the treatment group 3 (who received the task-adapted concept maps as scaffolding) performed significantly better than the control and treatment group 1. However, there was no significant difference between the groups who received task-adapted concept maps and the highlighted concept maps.

Chapter 7

Conclusion

Empirical studies have shown that concept maps are effective tools to facilitate meaningful learning. However, the widespread adoption of concept mapping as a technique is hindered by the substantial assistance and feedback required by learners to construct concept maps, and further, the additional workload involved to construct expert maps by teachers.

This thesis has explored ways of generating concept maps with minimal human intervention. In doing so, there have been several contributions to the field of research. The main contribution of this thesis is the development of a concept map mining framework (CMMF) to extract concept maps from lecture slides. Additionally, this thesis contributed by adopting the concept maps generated from CMMF as scaffolding resources to facilitate problem solving.

7.1 Main Contributions

The development of the CMMF contributes to several technical advancements in the field of research. Initially, in order to prepare the slide text for knowledge acquisition, the thesis proposed an approach to utilise contextual features in order to eliminate noisy data, pronoun resolution, replacing demonstrative determiners and transforming incomplete sentence fragments into complete sentences. A series of studies conducted to evaluate these approaches obtained promising results of 94%, 96% and 95% for precision, recall and F-measure respectively for noise detection. Further, an average of 64% for F-measure was obtained for pronoun resolution and compared well to the average inter-rater agreement of 67%.

CMMF is capable of automatically acquiring all important elements of concept maps (*concepts*, *relations* and *hierarchy*) in contrast to the other CMM systems which focused only on extracting some elements of the concept maps such as concepts and relations without relation labels and hierarchies (N.-S. Chen et al., 2008; S.-M. Chen & Bai, 2010; Clariana & Koul, 2004; Lee et al., 2009; S.-S. Tseng et al., 2007; Y.-H. Tseng et al., 2010), and concepts and relations without hierarchies (Alves et al., 2001; Olney, 2010; Olney et al., 2011; Valerio & Leake, 2006; Wang et al., 2008; Zouaq & Nkambou, 2008). Further, the creation of *cross-links* is possible when the produced CXL files are exported to the IHMC CMap tools for visualisation.

This thesis developed a full-scale *open relation* extraction technique using NLP-based algorithms and the arrangement of the concepts into a hierarchy with the use of natural layout of the slides. Although the triple extraction algorithm (*F-measure* of 69%) did not perform as well as humans (*inter-rater agreement* of 81%), results are promising with the combination of *well-written* as well as *ill-written* nature of text in lecture slides.

CMMF introduced a decision tree-based algorithm design to extract the knowledge in a sequence, ensuring the minimum loss of useful information. Initially, subject-verb-object (SVO) triples are extracted from *simple* sentences followed by SVO triple extraction from *complex* sentences. Afterwards, probable triple candidates are extracted from the rest of the sentences which have the correct order of nouns and verbs. Finally, the remaining texts are utilised to extract domain concepts with the aim of the creation of hierarchical relations using the natural layout of the lecture slides.

In addition to the contributions to the area of knowledge acquisition, the CMMF proposed an evaluation methodology to overcome the issues associated with binary classification system which evaluates the extracted information within educational context as *relevant* or *non-relevant*. The evaluation methodology proposed in this thesis considered the *relevance* measure of as a rank, with some highly important knowledge, averagely important knowledge and others with low importance.

In order to measure the effectiveness of concept maps extracted from lecture slides using CMMF, the correlation between computer-generated concept maps and human generated maps was calculated with the use of the evaluation methodology proposed in this thesis. Results were promising where some of the Computer Science courses obtained strong positive correlation ($r_s > 0.7$) with human maps, supporting the original hypothesis '*It is possible to develop auto-generated concept maps of lecture slides to strongly correlate with human constructed maps*'.

Overall, it can be stated that this thesis confirmed that the machine generated concept maps from lecture slides can be utilised as a positive alternative to expert maps. Thus, this thesis represents the first study, to the authors' knowledge, that presents new techniques to extract useful knowledge not only from well-written text but also ill-written text within Computer Science domain.

The choice of lecture slides as the knowledge source for concept map generation has potential benefits to both students and teachers. Lecture slides include little or no support for integrated knowledge organisation due to their point-based and linear nature (Brandt et al., 2001; Kinchin et al., 2008; Kinchin et al., 2000; Taber, 1994). Accordingly, concept map generation from lecture slides as a supplementary learning technique is important in order to support knowledge

organisation through meaningful learning and produce an epistemologically balanced teaching approach (All et al., 2003; Kinchin, 2006a).

However, one of the limitations of the CMMF is its *text-only* aspect. Currently, CMMF is not capable of extracting knowledge from *figures* or *tables*. It is challenging to extract triples from multimedia and tabular data. Tabular data contains combinations of lists of items and their descriptions or comparison texts. Therefore, it is difficult to develop algorithms to deal with the large variations of tabular patterns. Additionally, existing tools such as OCR do not provide reliable output when extracting text from images included in lecture slides due to their varied quality. Therefore, future work is necessary to develop algorithms to extract useful knowledge from *figures* and *tables* to reduce the information loss.

Lecture slides occasionally contain text in the *negative* form (e.g. *the class has no member function or member variable*) to emphasise disadvantages of knowledge components or describe bad habits of programming. CMMF's inability to correctly recognise some form of negations results producing incorrect triples. This limitation occurs due to a lack of an existing specific grammatical structure or 'tag' for negation. In future work, existing negation patterns should be learned from a suitable corpus in order to identify them in unseen data.

Some pronouns observed in this research included 'you, we, us, and itself' to address students who refer the course material or listening to the lecture. The context of a slide does not provide sufficient information to find the replacement candidates for the 'gender-related' pronouns. Therefore, CMMF excludes triple extraction from sentences that contain 'insolvable' pronouns (see Table 6.6). Gender-related pronouns were resolved in a different context by a work of Leskovec et al. (2004).

It is extremely difficult to find replacement candidates for some sentences with pronouns observed in the corpus. These generally include more than one dependent clause which make it difficult for even human interpretation (e.g. *need to write it to disk before replacing it if it was updated since it was last fetched from/written to disk*). In addition, some sentences include 'dummy' pronouns which do not contain a corresponding replacement in the context. Therefore, 'dummy' pronouns are usually excluded from processing (N. Ge et al., 1998). For instance, in the sentence '*it is raining*', the pronoun '*it*' adds no meaning to the sentence but it is required by syntax.

The concept maps generated using CMMF has the potential of being used by learners as scaffolding for problem solving and this thesis develops a framework (TASF) for that purpose. A research by Eylon & Reif (1984) provided a basis for introducing task-adapted knowledge

organisation techniques to provide more relevant information for solving problems. However, it is not a feasible approach to organise the knowledge according to tasks *manually*. Therefore, this thesis takes a new perspective to develop a machine-based framework for task adaptation. To the best of the authors' knowledge, this is the first study which facilitates problem solving with scaffolding via task-adapted knowledge organisation in an *automated manner*.

The results of utilising task-adapted concept maps as scaffolding for answering questions are promising with the task-adapted concept map group achieving higher post-test scores and obtaining statistically significant results compared to students who received lecture slides as scaffolding or full concept maps extracted using CMMF as scaffolding. However, there were no significant differences between the task-adapted concept map group and the group which received full concept maps with highlighted problem solving context.

TASF utilised triple extraction algorithms developed for CMMF to identify question triples in order to extract task-adapted concept maps. TASF has no restrictions on the grammatical structure of the question texts, in contrast to the other related works which have restrictions on the *grammatical structure* of the input questions (Dali et al., 2009; Lopez et al., 2007).

To simplify the automation, TASF restricts its question types into two categories known as 'descriptive' or 'comparison' (see Table 5.3). The 'question type' determines the amount of information included in the task-adapted concept map in terms of number of concepts and the proximity between matching nodes. It is possible that the context of task-adapted concept maps could be improved if the corresponding question types include more levels (see Table 2.3 and 2.4).

7.2 Future Directions

A number of future avenues are opened as a result of this thesis. Among them, the research can be extended from the development perspective to provide its benefits to a broader community. Currently, CMMF demonstrates its effectiveness within the Computer Science domain. As future research, the strengths and weaknesses of the CMMF can be further assessed using various domains such as Education, Psychology, Medicine, and Biology. This would help to propose a more accurate classification system to identify what type of course content is *well-fitted* to the framework. A cross-disciplinary framework will help to produce more robust concept maps and identify additional features to improve the set of algorithms.

Presently, the research is developed as a standalone system. As a future development, the research is expected to integrate to Learning Management Systems (LMS). This provides opportunities to embed this research into existing pedagogy. An option for teachers to 'generate

concept maps' from the lecture slides uploaded into a LMS will help to obtain the conceptual overview of the material. In addition, teachers can provide quizzes to a LMS and select suitable concept maps to produce task-adapted concept maps as scaffolding. Some existing Moodle plugins such as 'Concept map question type' by Villalon (2011) are suitable to improve for this purpose.

Algorithms developed in the current research can be altered for knowledge acquisition from various other education sources such as forum posts (Lau et al., 2008), multimedia resources (e.g. video lectures and podcast) and students' written answers to assessments. For instance, with the rapid growth of online fora for study discussion, the generation of concept maps using forum posts will provide analytics for teachers including what concepts are mostly discussed, which set of concepts discussed together, and further, sentiment and topic analysis can be performed to identify learners' misconceptions and knowledge gaps. This might help teachers to revise the generated concept maps, its basis lecture material or reflect on pedagogy to improve teaching strategies. Additionally, similar to students' essays as the basis material in the work of Villalon & Calvo (2011), concept map generation from scientific writing will provide opportunities for self-assessment through cognitive visualisation. In general, concept maps generated from various education sources can be merged to construct a consolidated domain concept map which could be expected to improve the pedagogical value of the map. The XML-based nature of the generated concept maps (i.e. CXL files) from this research allows for the integration of related concept maps constructed manually or extracted using other tools.

As per the suggestions of students' through the questionnaire, the concept maps generated using CMMF can be enhanced to embed resources to concepts including underlying lecture slides or external resources such as multimedia similar to the knowledge models discussed in the *Mars* project by NASA scientists (Canas, Carvalho, et al., 2004; Leake et al., 2003; Weinbrenner et al., 2011). These resources are suggested to provide more details of specific concepts.

In addition to the improvement of the current framework, the concept map mining research itself can be extended to utilise in other research areas. Presently, domain knowledge of intelligent educational systems (e.g. intelligent tutoring systems - ITS) is modelled manually by knowledge engineers with the use of domain experts or semi-automated authoring systems (Mitrovic, Koedinger, & Martin, 2003). This process takes from several months to years to complete. For instance, SQL-Tutor contains over 700 constraints, each taking over an hour to develop, resulting in more than four months to construct the knowledge base (Mitrović, 1998). However, a substantial problem with both manual and semi-automated processes is that the extensive effort and time dedicated to construct the knowledge base is not reusable due to

restrictions in the environment they are developed for (e.g. SQL-Tutor (Mitrovic et al., 2003), Algebra tutor (Corbett, Trask, Scarpinato, & Hadley, 1998), and Betty's Brain within the Biology domain (Leelawong & Biswas, 2008)). Therefore, the effort involved in domain modelling can be minimised when auto-generated concept maps are utilised as the first step to produce domain ontologies (Starr & Parente de Oliveira, 2013), and further gains when using domain independent knowledge sources (lecture slides) to produce intelligent tutoring systems for more than one domain. Therefore, concept maps can act as the preliminary step to produce domain ontologies as discussed in *TEXTCOMON* (Text-Concept map-Ontology) which extracts concept maps from text to transform into domain ontologies (Zouaq & Nkambou, 2008, 2009). In addition, some ITSs utilise expert maps to compare the students' maps such as in *Betty's Brain* (Leelawong & Biswas, 2008) and *CIRCSIM-Tutor* (Evens et al., 1997). Therefore, the concept maps generated from lecture slides can be utilised to compare with student maps within intelligent educational systems and provide feedbacks to students accordingly.

Further, concept maps as the domain model can be utilised to generate questions for students to acquire necessary skills. Question generation from concept maps in the Biology domain has already been discussed in the work of Olney et al. (2012). This process can be improved to introduce the adaptation to the educational environments such as providing adaptive learning material or designing adaptive learning paths to guide learners (Brusilovsky, 2004; Wolf, 2003). For instance, by identifying what students already know and what they are lacking to generate questions with varied complexity for individual students.

Apart from the extensions within the educational context, the notion of task-adapted knowledge organisation can be introduced to the information retrieval field, particularly within 'question answering' systems. In addition to returning the direct text answer for users' queries, the most relevant information to the query can be provided as a knowledge graph, allowing for parallel processing and contextualised question answering (Bradesko et al., 2010; Dali et al., 2009; Dali et al., 2010).

7.3 Concluding Remarks

The concept maps extracted using CMMF had a strong positive correlation with the human generated maps. Therefore, auto-generated concept maps can be utilised as a positive alternative to the manual construction of expert maps and further, it is possible to utilise the auto-generated concept maps for a wider range of applications within the educational context in the future. Additionally, the adoption of the task-adapted concept maps as scaffolding had a positive impact on students' learning and performance. Therefore, task-adapted concept maps can be utilised within the problem solving context to improve learning in the future.

Appendix A

Computer Science Course Outlines

This section provides outlines of Computer Science courses of University of Adelaide utilised for algorithm evaluation of CMMF. These courses are categorised according to the undergraduate levels.

Level 1:

COMP SCI 1101 - Introduction to Programming

Algorithms and problem-solving:	Problem-solving strategies; the role of algorithms in the problem-solving process; implementation strategies for algorithms; debugging strategies; the concept and properties of algorithms
Fundamental programming constructs:	Syntax and semantics of a higher-level language; variables, types, expressions, and assignment; simple I/O; conditional and iterative control structures; functions and parameter passing; structured decomposition
Fundamental data structures:	Primitive types; arrays; records; strings and string processing
Software development methodology:	Fundamental design concepts and principles; testing and debugging strategies; test-case design (black box testing and requirements testing); unit testing; programming environments
Human-computer interaction: Introduction to design issues - Social context of computing	History of computing and computers; evolution of ideas and machines; social impact of computers and the Internet; professionalism, codes of ethics, and responsible conduct; copyrights, intellectual property, and software piracy

COMP SCI 1102 - Object Oriented Programming

Object-oriented programming:	Object-oriented design; encapsulation and information-hiding; separation of behavior and implementation; classes, subclasses, and inheritance; polymorphism; class hierarchies
Fundamental computing algorithms:	simple searching and sorting algorithms (linear and binary search, selection and insertion sort)
Fundamentals of event-driven programming - Machine level representation of data:	Bits, bytes, and words; numeric data representation and number bases; representation of character data
Introduction to computer graphics: Using a simple graphics API - Memory management - Overview of programming languages:	History of programming languages; brief survey of programming paradigms
Introduction to language translation:	Comparison of interpreters and compilers; language translation phases; machine-dependent and machine-independent aspects of translation

COMP SCI 1103 - Algorithm Design & Data Structures

Review of elementary programming concepts - Fundamental data structures:	Stacks; queues; linked lists
Object-oriented programming:	Object-oriented design; encapsulation and information hiding; classes; separation of behaviour and implementation; class hierarchies; inheritance; polymorphism
Fundamental computing algorithms:	$O(N \log N)$ sorting algorithms;
Recursion:	The concept of recursion; recursive mathematical functions; simple recursive procedures; divide-and-conquer strategies; recursive backtracking; implementation of recursion
Basic algorithmic analysis:	Asymptotic analysis of upper and average complexity bounds; identifying differences among best, average, and worst case behaviours; big "O," little "o," omega, and theta notation;
Algorithmic strategies:	Brute-force algorithms; greedy algorithms; divide-and-conquer; backtracking; branch-and-bound; heuristics; pattern matching and string/text algorithms; numerical approximation algorithms
Overview of programming languages: Programming paradigms - Software engineering:	Software validation; testing fundamentals, including test plan creation and test case generation; object-oriented testing
Software evolution:	Software maintenance; characteristics of maintainable software; reengineering; legacy systems; software reuse

Level 2:

COMP SCI 2006: Introduction to Software Engineering

Design:	software design, UML notation, static models - identifying classes and associations, dynamic models - identifying states, events, transitions, use cases, mapping designs into code
Specification:	the scope, role and styles of software specification
Testing:	modes of testing, organising test suites
Human issues:	managing object-oriented projects, ethics, professional practice

Level 3:

COMP SCI 3004 - Operating Systems

OS purposes:	resource management and the extended virtual computer; historical development.
Processes:	critical sections and mutual exclusion, semaphores, monitors, classical problems, deadlock; process scheduling
Input and Output:	hardware and software control
Memory management:	multi-programming; swapping; virtual memory, paging and symbolic segmentation;
File System:	operations, implementation, performance
Protection mechanisms:	protection domains, access lists, capability systems, principle of minimum privilege

COMP SCI 3012 – Distributed Systems

the challenges faced in constructing client/server software:	partial system failures, multiple address spaces, absence of a single clock, latency of communication, heterogeneity, absence of a trusted operating system, system management, binding and naming
Techniques for meeting these challenges:	RPC and middleware, naming and directory services, distributed transaction processing, 'thin' clients, data replication, cryptographic security, mobile code. Introduction to Java RMI

Level 4:

COMP SCI 4000 – Software Architecture

Fundamental principles and guidelines for software architecture design, architectural styles, patterns and frameworks
Methods, techniques and tools for describing software architecture and documenting design rationale
Software architecture design and evaluation processes
Rationale and architectural knowledge management in software architecting
Approaches and tools for designing and evaluating software architectures for the state of the art technologies such as cloud-computing and service-operation and mobile computing
Future challenges and emerging trends in software architecture discipline

Apart from these courses, there are other CS courses within University of Adelaide utilised for manual analysis to develop features and algorithms.

Undergraduate year	Course
One	COMP SCI 1007 – Computer Science Concepts*
Two	COMP SCI 2000 - Computer Systems
	COMP SCI 2002 – Database and Information Systems
	COMP SCI 2201 – Algorithm and Data Structure Analysis
	COMP SCI 2202 - Foundations of Computer Science
	COMP SCI 2202B - Foundations of Computer Science B
Three	COMP SCI 3001 – Computer Networks & Applications
	COMP SCI 3002 – Programming Techniques*
	COMP SCI 3005 – Computer Architecture
	COMP SCI 3007 – Artificial Intelligence
	COMP SCI 3009 – Advanced Programming Paradigms
	COMP SCI 3013 – Event Driven Computing
Four	COMP SCI 4023 – Software Process Improvement
	COMP SCI 4054 – High Integrity Software Engineering
Higher levels	COMP SCI 7045 – Distributed and High-Performance Computing
	COMP SCI 7092 – Mobile and Wireless Networks

*Courses not currently offered

In addition, some lecture slide sets from text book publishers utilised for algorithm analysis.

CS Course	Outline	Web link to resources
Software Engineering	Introduction, Software Processes, Agile Software Development, Requirements Engineering, System modelling, Architectural design, Design and implementation, Software testing, Software evolution, Socio-technical systems, Dependability and Security, Dependability and Security specification, Dependability Engineering, Security Engineering, Dependability and Security Assurance, Software reuse, Component-based software engineering, Distributed software engineering, Service-oriented architecture, Embedded systems, Aspect-oriented software engineering, Project Management, Project planning, Quality management, Configuration management, Process improvement	http://ifs.host.cs.st-andrews.ac.uk/Books/SE9/Presentations/ (Sommerville, 2010)
Computer Networking	Introduction, Application layer, Transport layer, Network layer, Link layer, Wireless and Mobile networks, Multimedia networking, Security, Network management	http://www-net.cs.umass.edu/kurose-ross-ppt-6e/ (Kurose & Ross, 2012)
Operating Systems	Overview: Introduction, Operating system structures; Process Management: Processes, Threads, CPU scheduling, Process synchronization, deadlocks; Memory management: main memory, virtual memory; Storage Management: file-system interface, file-system implementation, mass-storage structure, I/O systems; Protection and security: Protection, security; Distributed systems: distributed system structures, distributed file systems, distributed coordination; special-purpose systems: real-time systems, multimedia systems; Case studies: Linux system, Windows XP, Historical perspective	http://codex.cs.yale.edu/avi/os-book/OS8/os8c/slide-dir/ (Silberchatz et al., 2012)
Data and Computer Communications	Introduction, Protocols and architecture, Data transmission, Transmission media, Data encoding, Data communication interface, Data link control, Multiplexing, Circuit switching, Packet switching, ATM and Frame Relay, Congestion in data networks, LAN Technology, LAN Systems, Internetwork protocols, Internetwork operation, Transport protocols, Network security, Distributed applications	http://williamstallings.com/DCC6e.html (Stalling, 2007)

Appendix B

Algorithms and Java Implementations

Algorithm B.1: Split verb phrases by CC to allocate subjects

Require: VP subtree // Tree annotation of verb phrase

1. If VP subtree contains the pattern 'VP CC VP'
2. Array of VP subtree <- split by 'CC'
3. For each VP subtree in array of sub trees
 Array of sentences <- subject U VP sub tree
1. For each sentence in array of sentences
 Extract triples

Output: complete sentences

Algorithm B.2: Concept extraction using Greedy approach

Require: sentence // input sentence text
 : part_of_speech // part of speech tagging of sentence
 : regular patterns // hand-written regular expression
patterns

1. For each regular pattern in the array
2. Match regular pattern with part_of_speech of sentence
3. while match found
 - a. Get 'length' of matched pattern using no of tokens
 - b. Get 'index' of matching pattern in part-of-speech
 - c. Extract corresponding 'key term' in sentence using length and index
 - d. Remove 'key term' from sentence and put it into a list
- Repeat until sentence is empty
4. End for each

Output: list of key terms

Algorithm B.3: Probability calculation for font face 'Courier new'

Require: rich text run // text with same style

1. Get the font face of rich text run and store in a map with the frequency of each face occurs
2. Calculate total frequency as total
3. Probability for 'courier new' = frequency / total

Output: probability of typographic feature

Algorithm B.4: Concept map merging

```
Require: CXL 1          // first CXL file to merge
           CXL 2          // second CXL file to merge

1. Obtain root id of CXL1
2. Read ids of concepts in the concept-list of CXL1
3. If root id equals to a id in the concept list
   a. Obtain label of corresponding concept as 'root label'
   b. Read concept labels in the concept-list of CXL2
   c. If match found
       i. Obtain concept label as 'concept label'
       d. End if
   End if
4. If 'root label' equals to 'concept label'
   a. Obtain id of 'concept label' as 'concept id'
   b. Replace root id in CXL1 with 'concept id' in all
       records
   c. Put concept-list, linking-phrase-list and connection-
       list of CXL1 to CXL2
   End if

Output: merged CXL file
```

Algorithm B.5: CXL to TreeGraph converter

```
Require: id // root id of CXL file
           concept list // concept list of CXL file
           connection list // connection list of CXL file
           linking phrase list // linking phrase list of CXL
file

1. For each concept in concept list
  2a Obtain start label from concept list using id
  2b Set label of TreeGraphNode
  2c Obtain 'to' connections of id from connection list
  2d For each 'to' connection
    i. Obtain edge labels from linking phrase list
    ii. Set edge label of TreeGraphEdge
    iii. Set TreeGraphEdge of TreeGraphNode
    iv. Obtain 'from' connections
        v. For each 'from' connection in connection list
            Obtain labels of children from concept
            list
            Set children label of Children
            Set children of TreeGraphNode
            Set concept as key (parent) and children
            list as values (children list) in a
            hashmap
            If concept in the hashmap as values
                Set parent list of TreeGraphNode
            Else
                // root node
                Set parent list of TreeGraphNode as
                null
            End if
        vi. End for each
  2e Add TreeGraphNode to TreeGraph
  2f Set root of TreeGraph
  2g End for each
2. End for each

Output: TreeGraph // Converted TreeGraph from CXL file
```

Java implementation of Synonym extraction from WordNet

```
System.setProperty("wordnet.database.dir", "C://Program
Files/WordNet/2.1/dict")
WordNetDatabase database = WordNetDatabase.getFileInstance()

Synset[] synsets = database.getSynsets(word) //synonyms

if(synsets.length > 0)
  for(Synset s : synsets)
    String[] synonyms = s.getWordForms()
  end for
end if
```

Appendix C

Resources of In-class Experiments

This research is approved by the Human Research Ethics Committee of University of Adelaide (Project approval no: **H-2014-052**, ethics expiry date: 31 March 2017)

C1: Quizzes used for the main study

1. Development testing is a stage within Software engineering life cycle which includes all testing activities carried out by the development team. Identify other testing stages
 - i. User testing
 - ii. Release testing
 - iii. System testing
 - iv. Acceptance testing
 - v. Regression testing

2. In component testing, errors are detected in different types of interfaces between program components. Identify the types of interface testing.
 - i. Procedural interface
 - ii. Application programming interface
 - iii. Parameter interface
 - iv. Message passing interface
 - v. Shared memory interface

3. In the context of Software testing, select statement(s) that best describes the purpose of Regression testing.
 - i. Ensures that the new testing environment (e.g. user site) have not created problems for the software
 - ii. Ensures that the interface errors have not created problems to the working code
 - iii. Ensures that the software operates properly on its maximum design load
 - iv. Ensures that newly added code have not created problems with previous versions of the program
 - v. Ensures that the newly added hardware have not created problems with previous version of the software

4. In the context of Software testing, define what is/are meant by Test-Driven Development (TDD)?
 - i. An approach where codes are written before tests
 - ii. An approach where tests are written before codes
 - iii. An approach where tests are carried out by separate testing team
 - iv. An approach to program development in which you combine testing and code development by alternating between them
 - v. An approach where one person involved in developing code and another person perform tests at the same time

5. A program validates user input as follows:
Value < 10 are rejected
Value between 10 and 21 are accepted
Value > = 22 are rejected

Which of the following input value set(s) covers all of the equivalence partitions?

- i. 10, 11, 21
 - ii. 3,20,21
 - iii. 3,10,22
 - iv. 10,21,22
 - v. 3,10,215
6. You develop a system that is designed to process up to 300 transactions per second. You start testing the system with less than 300 transactions per second and gradually increase the load until the system well beyond the maximum load (overload) and fail. Which testing type(s) suitable for this?
- i. Partition testing
 - ii. Performance testing
 - iii. Boundary testing
 - iv. Stress testing
 - v. Load testing
7. Compare and contrast system testing and release testing
- i. The team that involved in system testing is responsible for release testing
 - ii. Release testing is a form of system testing
 - iii. System testing should focus discovering bugs in the system while release testing checks that the system meets its requirements
 - iv. Both system and release testing are part of development testing
 - v. Both release testing and system testing are a black-box testing process
8. Compare and contrast black-box testing and white-box testing
- i. Black-box testing validates nested loops in the program while white-box testing validates simple loops in the program
 - ii. Black-box testing develops tests from hidden errors of the system while white-box testing develops tests from visible errors of the system
 - iii. Both black-box testing and white-box testing are part of release testing process
 - iv. Black-box testing develops tests from system implementation while white-box testing develops tests from system specification
 - v. Black-box testing develops tests from system specification while white-box testing develops tests from system implementation
9. Read the following simple requirement for an Automated Teller machine (ATM) program. "ATM only allows bank cards from ANZ bank. Customers are allowed cash withdrawal only and it should be between \$20-\$1000. They can receive only multiples of \$20 or \$50." Decide which of the following test case(s) pass from the requirement-based testing.
- i. Customer withdraws \$40 from ATM
 - ii. Customer wants to withdraw \$10 from ATM
 - iii. Customer inserts bank card from Common wealth bank
 - iv. Customer wants to withdraw money from ATM
 - v. Customer withdraws \$1500 from ATM
10. "It is not necessary for a program to be completely free of defects before it is delivered to its customers". What reason(s) do you think are valid to negotiate?
- i. Customers do not have technical knowledge, therefore they will not find defects
 - ii. Customers want to deploy the system for immediate benefits
 - iii. Contract between customer and the software company allow the system with bugs

- iv. Some bugs cannot be completely removed. Therefore, it is acceptable to deliver the system with bugs
- v. Cost of not using the new system may be greater than cost of working with defects

C2: Quizzes used for the post-test

Student ID

Please select or write answers to all questions

1. Release testing is a stage within Software Engineering life cycle which includes a separate team to test a complete version of the system before it is released to the users.
What are the other two testing stages?
.....
.....
2. A web-based library system is fully tested for 'book borrowing/returning' functionalities. The librarian requested a new functionality to 'renew books'. This requires setting a new 'returning date' by calculating number of factors such as whether the user exceeds maximum number of renewals, status of the user as 'undergrad, postgrad or staff'.
Which of the following testing type(s) ensures that the new 'renewal' function does not produce errors in the 'borrowing/returning' functionalities?
 - i. Stress testing
 - ii. Regression testing
 - iii. Black-box testing
 - iv. Performance testing
 - v. Partition testing
3. Test-driven development (TDD) is a program development approach where you first write before the
4. A program validates user input as follows:
"Value < 10 are rejected
Value between 10 and 21 are accepted
Value > = 22 are rejected"
What are the possible input values that cover all the equivalence partitions?
.....
5. An airline ticket reservation system allows 60 concurrent transactions per second. Due to an unexpected rush of travellers, 85 users attempted to book through the reservation system at the same time. This result in the reservation system being overloaded and crashed. Which type of testing is suitable in order to avoid such situations?
 - i. Load testing
 - ii. Performance testing
 - iii. Crash testing
 - iv. Stress testing
 - v. Regression testing
6. In component testing, errors are detected in different types of interfaces between program components. Identify the types of interface testing.
 - i. Procedural interface
 - ii. Parameter interface
 - iii. Application programming interface
 - iv. Message passing interface
 - v. Shared memory interface
7. Black-box testing develops tests from system specification while white-box testing develops tests from

8. “An ATM only allows bank cards from Commonwealth bank. Customers are allowed cash withdrawal only and it should be between \$50-\$1000. They can receive only multiples of \$50.” From the given requirements, please specify three test cases for ‘requirement-based testing’.

.....
.....
.....

9. Compare and contrast system testing and release testing
- i. Both the system testing and release testing are conducted by the same system development team
 - ii. System testing focuses on discovering defects while release testing checks that the system meets its requirements
 - iii. Both system testing and release testing are part of development testing
 - iv. Release testing is a form of system testing
 - v. Separate team that has not been involved in the system development is responsible for release testing

10. Explain why it is not necessary for a program to be completely free of defects before it is delivered to customers?

.....

C3: Sample screen shots of scaffolding resources

Question 1: *In component testing, errors are detected in different types of interfaces between program components. Identify the types of interface testing.*

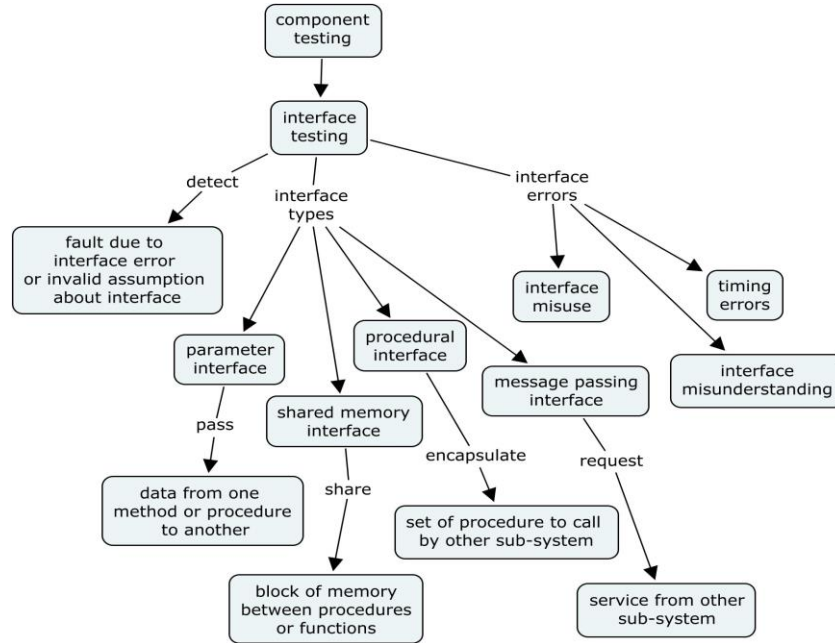


Figure C.1: Task-adapted concept map as scaffolding for question 1

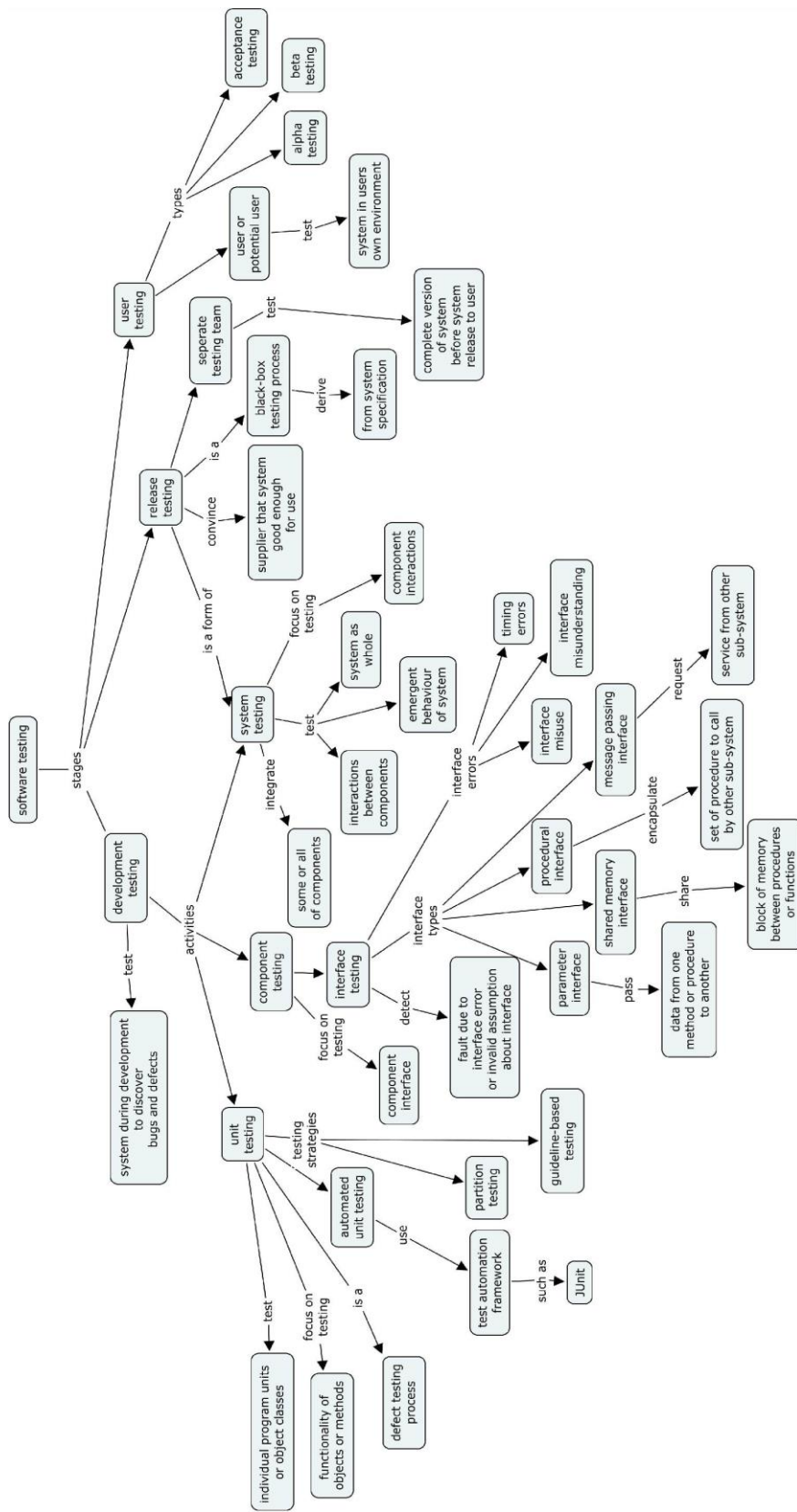


Figure C.2: Concept map as scaffolding for question 1

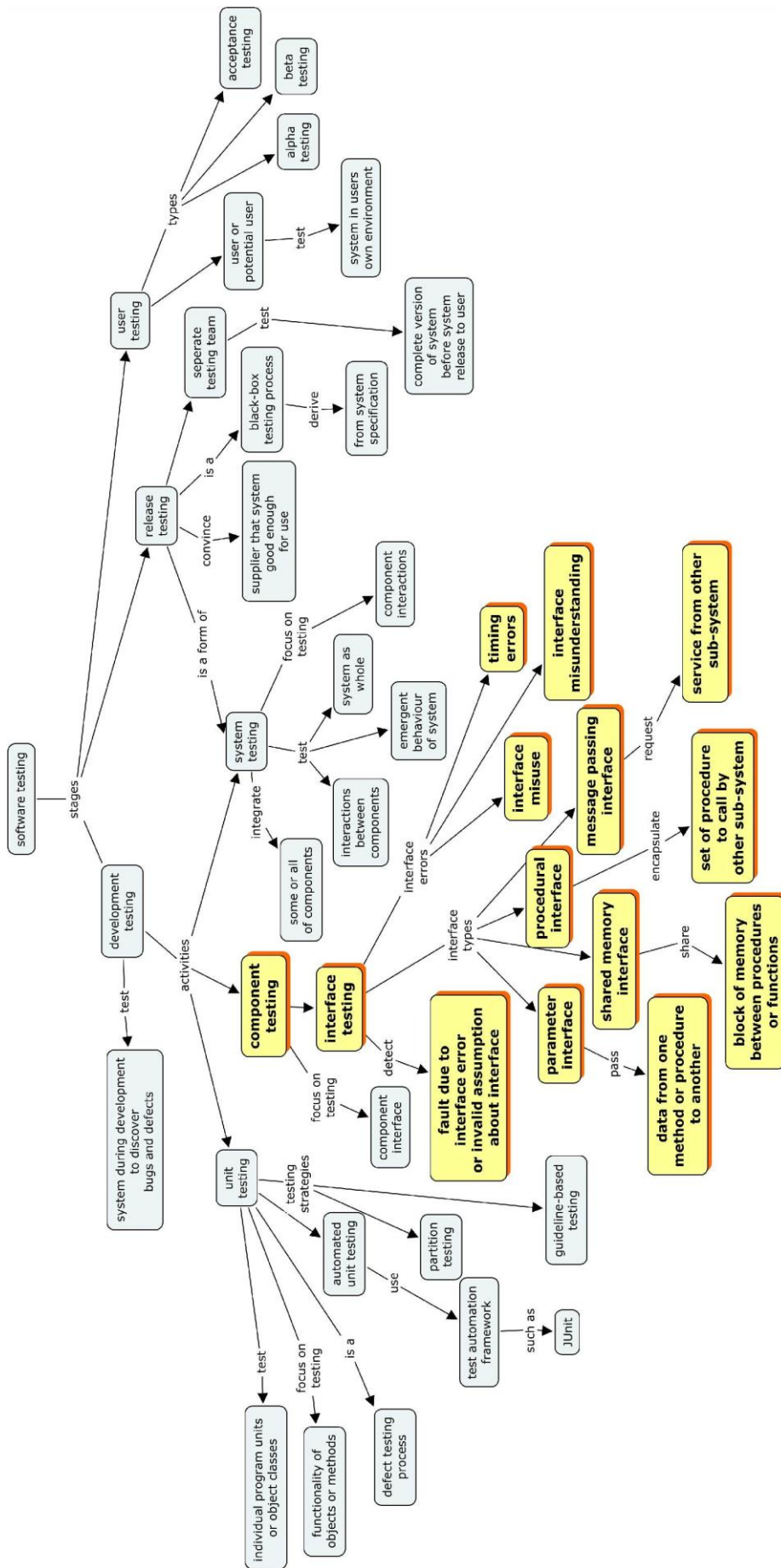


Figure C.3: Concept map with highlighted problem solving context as scaffolding for question 1

Question 2: Compare and contrast system testing and release testing.

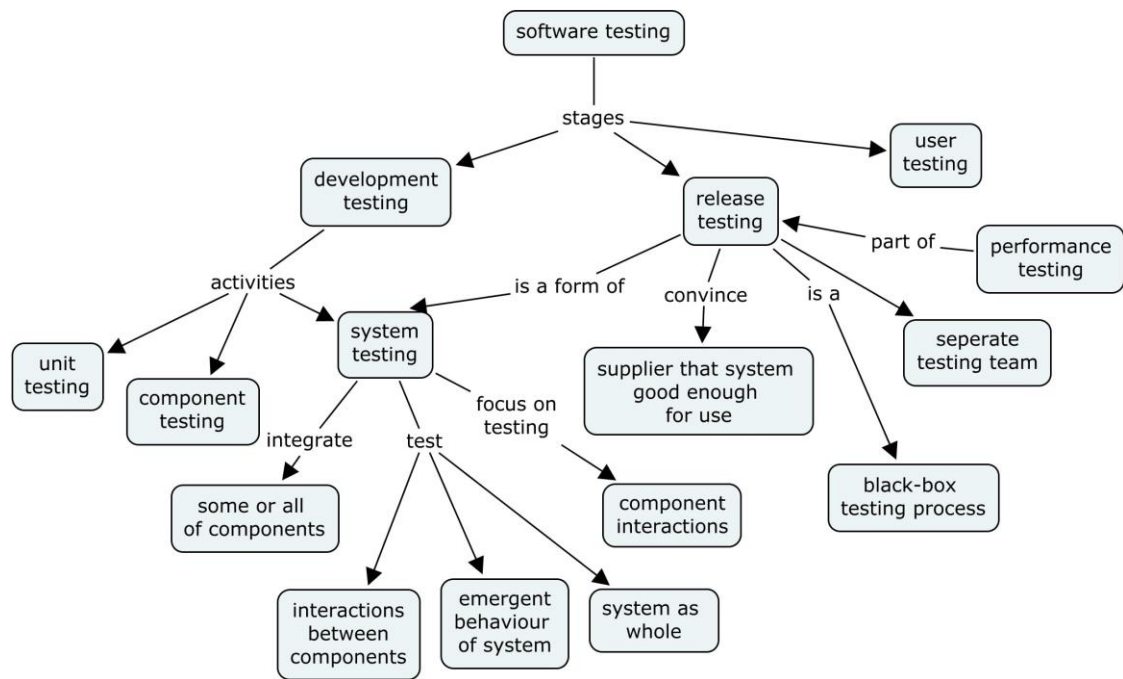


Figure C.4: Task-adapted concept map as scaffolding for question 2

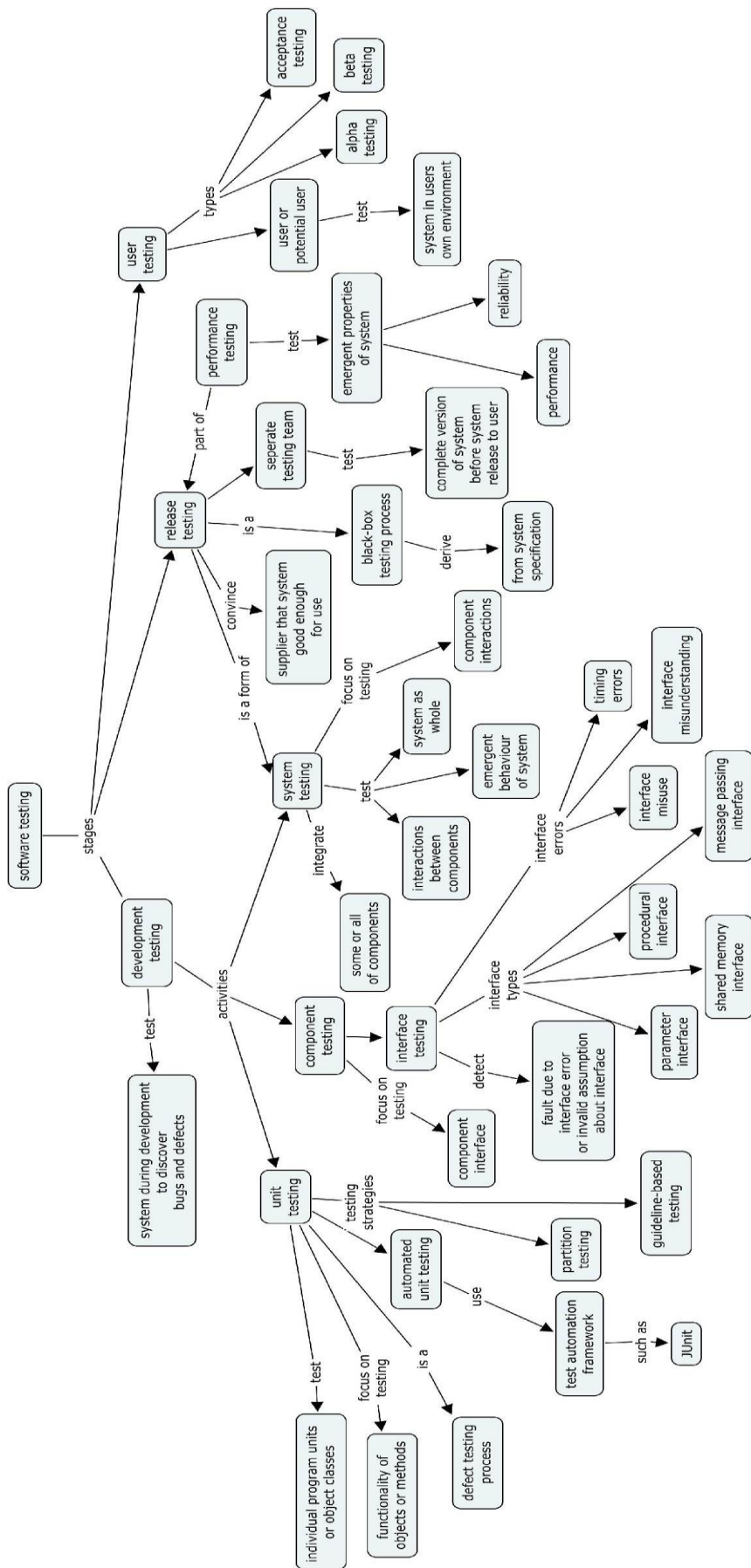


Figure C.5: Concept map as scaffolding for question 2

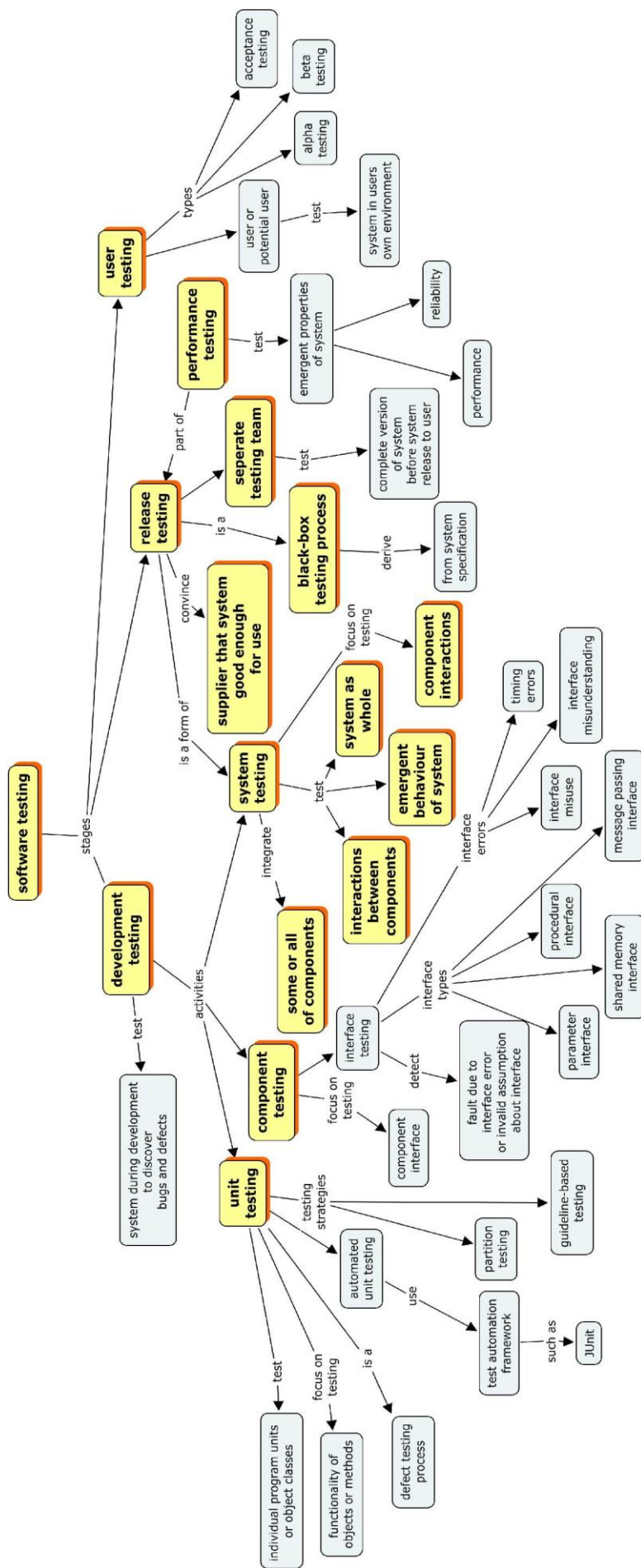


Figure C.6: Concept map with highlighted problem solving context as scaffolding for question 2

C4: Student questionnaire

Please circle or underline the answer

1. Have you used any knowledge organisation techniques before? (e.g. Concept maps, mind maps, semantic networks, ontology, knowledge maps)

No	I have heard, but never used	Yes, to some extent	Yes, I have used in the past	Yes, I am currently using
----	------------------------------	---------------------	------------------------------	---------------------------

2. If yes to Q1, please list the techniques below

3. What do you think about concept maps used in this study to answer questions?

Very helpful
Helpful to some extent
Neutral
Difficult to some extent
Very difficult

4. If it was difficult to use concept maps in this study, please select the statement (s) that describes your problem

None
I have problems in identifying concepts
I have problems in identifying relationships
I have problems in identifying structure or hierarchy
I have problems in identifying everything

5. In this research, we developed a tool to automatically generate concept maps from lecture notes. "This kind of tool will help answering questions".

Strongly disagree	Disagree	Neutral	Agree	Strongly agree
-------------------	----------	---------	-------	----------------

6. If our tool made available through CS forums in future, which resources do you prefer for your studies?

I would continue with my usual resources e.g. lecture notes
I would like to use concept maps generated by the system using lecture notes
I would like to use both lecture notes and concept maps
I would like to use partial concept maps that contain information related to answer each question
I would like to draw my own concept maps

7. Sometimes full concept maps generated by the system can contain lot of information. Therefore, "I would like a tool which can extract partial concept maps to assist answering questions"

Strongly disagree	Disagree	Neutral	Agree	Strongly agree
-------------------	----------	---------	-------	----------------

8. What type of courses do you think this kind of tool will be more useful?

Courses with less programming component such as Software Engineering
Courses with high programming component such as OOP, Introductory programming, ADDS
Every course
Not a single course
Please specify any other courses.....

9. Please write down any problems you had while interacting with the web-based system.

--

10. What are the features you would like to suggest to improve our tool?

--

C5: Student instruction sheet

Instruction Sheet

- You need to read and sign the **Consent form** before you start the tasks.

Guidelines to use the experimental system

1. Type the URL given on the whiteboard in Firefox web browser
 2. Log in to the system by using your student ID (e.g. a1234567)
 3. You will receive 10 Multiple choice questions (MCQ) to answer
(Note: some questions can have more than one correct answer)
 4. If you correctly answer the quiz, you will be automatically redirected to the next question
 5. If you have done a mistake, you will be prompted to re-attempt the same question.
At this stage, you can get **HELP** from the system by clicking 'help'
 6. If you are not familiar with concept maps, please refer to '**Concept map learning guide**'
 7. If you have any issues, please raise your hand and a supervisor will help you
- When you finish the practice test, please fill the simple **questionnaire**
 - Handover the questionnaire and consent form to a supervisor
 - Collect \$10 gift

Thank you for your participation!

Thushari Atapattu (PhD student)

Email: thushari@cs.adelaide.edu.au

C6: Example concept map for treatment groups

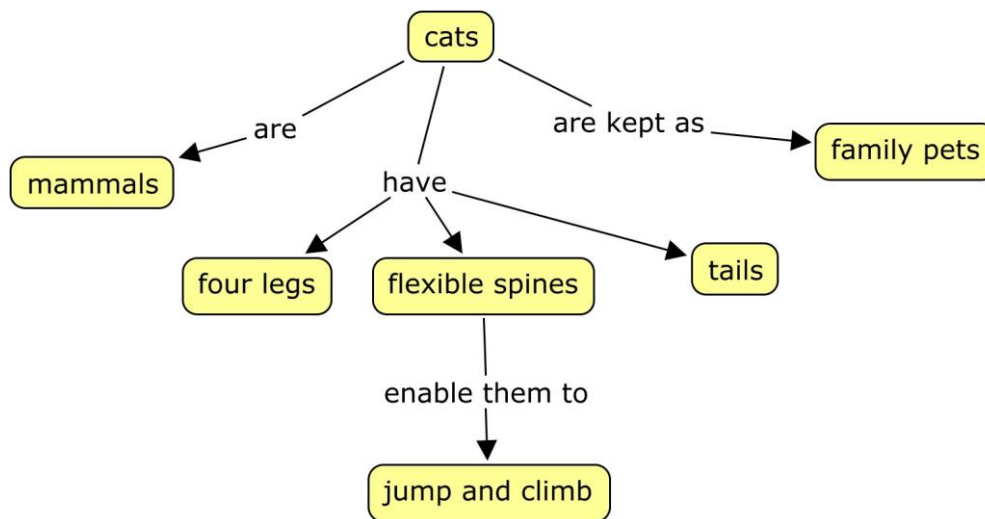
Concept map learning guide

What is a concept map?

Concept map is an educational tool to visualise knowledge. Concept map generally includes following elements:

1. Concepts - using ovals or boxes
2. Relationships - using links
3. Relationship labels - using linking words

Following figure shows an example concept map about 'cats'



This information is sufficient to understand the concept maps in this experiment.
If you have any issues, please ask from a supervisor

If you would like to learn more about concept maps later, please visit
<http://cmap.ihmc.us/docs/conceptmap.html>

Appendix D

Statistics of responses to the Questionnaire

Table D.1: What do you think about concept maps used in this study to answer questions?

Group	LS (%) (N = 10)	CMap (%) (N = 15)	HLCMap (%) (N = 14)	TSKCMMap (%) (N = 17)
Very helpful (1)	10	6.7	28.6	17.6
Helpful (2)	40	66.7	57.1	58.8
Neutral (3)	20	20	14.3	23.5
Difficult (4)	30	6.7	0	0
Very Difficult (5)	0	0	0	0

Table D.2: What are the issues related to concept maps in this study?

Group	CMap (%) (N = 15)	HLCMap (%) (N = 13)	TSKCMMap (%) (N = 15)
None (1)	26.7	38.5	73.3
Concepts (2)	20	53.8	0
Relationships (3)	13.3	0	20
Hierarchy (4)	40	7.7	6.7
Everything (5)	0	0	0

Table D.3: Do you think a tool to automatically generate concept maps from lecture slides will help answering questions?

Group	LS (%) (N = 10)	CMap (%) (N = 15)	HLCMap (%) (N = 13)	TSKCMMap (%) (N = 17)
Strongly disagree (1)	0	0	0	0
Disagree (2)	0	13.3	30.8	0
Neutral (3)	0	13.3	7.7	23.5
Agree (4)	100	46.7	23.1	58.8
Strongly agree (5)	0	26.7	38.5	17.6

Table D.4: Which form of resources do you prefer if our tool made available through CS Forums in future?

Group	LS (%) (N = 10)	CMap (%) (N = 15)	HLCMap (%) (N = 13)	TSKCMMap (%) (N = 17)
Continue to use lecture notes (1)	0	6.7	23.1	5.9
Concept maps generated from lecture notes (2)	10	6.7	0	5.9
Both lecture notes and concept maps (3)	80	80	53.8	82.4
Task-adapted concept maps (4)	10	6.7	23.1	5.9
Draw my own concept maps (5)	0	0	0	0

Table D.5: Would you like a tool which can extract partial concept maps to assist answering questions?

Group	LS (%) (N = 10)	CMap (%) (N = 15)	HLCMap (%) (N = 12)	TSKCMMap (%) (N = 17)
Strongly disagree (1)	0	0	0	0
Disagree (2)	0	0	8.3	0
Neutral (3)	10	20	25	29.4
Agree (4)	70	46.7	41.7	47.1
Strongly Agree (5)	10	33.3	25	23.5
Not Applicable (6)	10	-	-	-

Table D.6: What type of courses do you think this kind of tool will be more useful?

Group	LS (%) (N = 9)	CMap (%) (N = 12)	HLCMap (%) (N = 10)	TSKCMMap (%) (N = 14)
Less programming (e.g. Software Engineering)	44.4	41.7	70	14.3
More programming (e.g. IP, OOP, ADDS)	11.1	8.3	0	7.1
Every course	44.4	41.7	20	64.3
Not a single course	0	8.3	0	7.1
Specific courses (e.g. Electrical Engineering 1A)	0	0	10	7.1

References

- ACM. (2012). The ACM Computing Classification System (CCS). 2012, from http://dl.acm.org/ccs_flat.cfm#toc
- Afantenos, S., Karkaletsis, V., & Stamatopoulos, P. (2005). Summarization from medical documents: a survey. *Artif. Intell. Med.*, 33(2), 157-177. doi: 10.1016/j.artmed.2004.07.017
- Agichtein, E., & Gravano, L. (2000). *Snowball: extracting relations from large plain-text collections*. Paper presented at the Proceedings of the fifth ACM conference on Digital libraries, San Antonio, Texas, USA.
- Alibali, M. (2006). Does visual scaffolding facilitate students' mathematics learning? Evidence from early Algebra. University of Wisconsin, Madison.
- All, A. C., Huycke, L. I., & Fisher, M. J. (2003). Instructional tools for nursing education: concept maps. *Nurs Educ Perspect*, 24(6), 311-317.
- Alves, A. O., Pereira, F. C., & Cardoso, A. (2001). *Automatic Reading and Learning from Text*. Paper presented at the International Symposium on Artificial Intelligence Fort Panhala, India.
- Anderson, J. R. (1974). Retrieval of propositional information from long-term memory. *Cognitive Psychology*, 6(4), 451-474. doi: [http://dx.doi.org/10.1016/0010-0285\(74\)90021-8](http://dx.doi.org/10.1016/0010-0285(74)90021-8)
- Apache. (2011). Apache POI - the Java API for Microsoft Document. 2011, from <http://poi.apache.org/>
- Apperson, J. M., Laws, E. L., & Scepansky, J. A. (2008). An assessment of student preferences for PowerPoint presentation structure in undergraduate courses. *Computers & Education*, 50(1), 148-153. doi: <http://dx.doi.org/10.1016/j.compedu.2006.04.003>
- Atapattu, T., Falkner, K., & Falkner, N. (2012). Automated Extraction of Semantic Concepts from Semi-structured Data: Supporting Computer-Based Education through the Analysis of Lecture Notes. In S. Liddle, K.-D. Schewe, A. Tjoa & X. Zhou (Eds.), *Database and Expert Systems Applications* (Vol. 7446, pp. 161-175): Springer Berlin Heidelberg.
- Atapattu, T., Falkner, K., & Falkner, N. (2014a). *Acquisition of Triples of Knowledge from Lecture Notes: A Natural Language Processing Approach*. Paper presented at the The 7th International Conference on Educational Data Mining, London, United Kingdom.
- Atapattu, T., Falkner, K., & Falkner, N. (2014b). *Evaluation of Concept Importance in Concept Maps Mined from Lecture Notes: Computer vs Human*. Paper presented at the The 6th International Conference on Computer Supported Education, Barcelona, Spain.
- Atkins-Sayre, W., Hopkins, S., Mohundro, S., & Sayre, W. (1998). Rewards and Liabilities of Presentation Software as an Ancillary Tool : Prison or Paradise? . New York, NY: Distributed by ERIC Clearinghouse.
- Ausubel, D. (1968). *Educational Psychology: A Cognitive View*. New York and Toronto: Holt, Rinehart and Winston.
- Ausubel, D. (2000). *The Acquisition and Retention of Knowledge: a Cognitive View*. Dordrecht, Boston: Kluwer Academic Publishers.
- Ausubel, D., & Robinson, F. G. (1969). *School Learning: An Introduction to Educational Psychology*. NY: Holt, Rinehart & Winston Inc.
- Azevedo, R., Cromley, J. G., & Seibert, D. (2004). Does adaptive scaffolding facilitate students' ability to regulate their learning with hypermedia? *Contemporary Educational Psychology*, 29(3), 344-370. doi: <http://dx.doi.org/10.1016/j.cedpsych.2003.09.002>
- Bach, N., & Badaskar, S. (2007). *A review of relation extraction*. Paper presented at the Literature review for language and statistics II, Carnegie Mellon University.
- Bagno, E., & Eylon, B.-S. (1997). From problem solving to a knowledge structure: An example from the domain of electromagnetism. *American Journal of Physics*, 65(8), 726-736. doi: <http://dx.doi.org/10.1119/1.18642>

- Banko, M., Cafarella, M. J., Soderland, S., Broadhead, M., & Etzioni, O. (2007). *Open information extraction from the web*. Paper presented at the Proceedings of the 20th international joint conference on Artificial intelligence, Hyderabad, India.
- Bartsch, R. A., & Cobern, K. M. (2003). Effectiveness of PowerPoint presentations in lectures. *Computers & Education*, 41(1), 77-86. doi: 10.1016/s0360-1315(03)00027-7
- Bascones, J., Venezuela, & Novak, J. D. (1985). Alternative instructional systems and the development of problem-solving skills in physics. *European Journal of Science Education*, 7(3), 253-261. doi: 10.1080/0140528850070304
- Basque, J., & Lavoie, M.-c. (2006). *Collaborative Concept Mapping in Education: Major Research Trends*. Paper presented at the Second International Conference on Concept Mapping.
- Bean, T. W., & Stevens, L. P. (2002). Scaffolding Reflection for Preservice and Inservice Teachers. *Reflective Practice*, 3(2), 205-218. doi: 10.1080/14623940220142343
- Beed, P. L., Hawkins, E. M., & Roller, C. M. (1991). Moving Learners toward Independence: The Power of Scaffolded Instruction. *The Reading Teacher*, 44(9), 648-655. doi: 10.2307/20200767
- Bloom, B., Englehart, M., Furst, E., Hill, W., & Krathwohl, D. (1956). *Taxonomy of educational objectives: The classification of educational goals. Handbook I: Cognitive domain*. New York, Toronto: Longmans, Green.
- Bower, G. H., Clark, M. C., Lesgold, A. M., & Winzenz, D. (1969). Hierarchical retrieval schemes in recall of categorized word lists. *Journal of Verbal Learning and Verbal Behavior*, 8(3), 323-343. doi: [http://dx.doi.org/10.1016/S0022-5371\(69\)80124-6](http://dx.doi.org/10.1016/S0022-5371(69)80124-6)
- Boxtel, C. v., Linden, J. v. d., & Kanselaar, G. (2000). Collaborative learning tasks and the elaboration of conceptual knowledge. *Learning and Instruction*, 10(4), 311-330. doi: [http://dx.doi.org/10.1016/S0959-4752\(00\)00002-5](http://dx.doi.org/10.1016/S0959-4752(00)00002-5)
- Bradesko, L., Dali, L., Fortuna, B., Grobelnik, M., Mladenic, D., Novalija, I., & Pajntar, B. (2010, 21-24 June 2010). *Contextualized question answering*. Paper presented at the Information Technology Interfaces (ITI), 2010 32nd International Conference on.
- Bradley, J. H., Paul, R., & Seeman, E. (2006). Analyzing the structure of expert knowledge. *Inf. Manage.*, 43(1), 77-91. doi: 10.1016/j.im.2004.11.009
- Brandt, L., Elen, J., Helleman, J., Heerman, L., Couwenberg, I., Volckaert, L., & Morisse, H. (2001). The impact of concept mapping and visualization on the learning of secondary school chemistry students. *International Journal of Science Education*, 23(12), 1303-1313. doi: 10.1080/09500690110049088
- Brank, J., Grobelnik, M., & Mladenic, D. (2005). *A survey of ontology evaluation techniques*. Paper presented at the Data Mining and Data Warehouses (SiKDD).
- Brin, S. (1999). *Extracting Patterns and Relations from the World Wide Web*. Paper presented at the International Workshop on The World Wide Web and Databases.
- Brusilovsky, P. (2004). *KnowledgeTree: a distributed architecture for adaptive e-learning*. Paper presented at the Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters, New York, NY, USA.
- Bunescu, R. C., & Mooney, R. J. (2005). *A shortest path dependency kernel for relation extraction*. Paper presented at the Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing, Vancouver, British Columbia, Canada.
- Calvo, I., Arruarte, A., Elorriaga, J. A., Larranaga, M., & Conde, A. (2011, 12-15 Oct. 2011). *The use of concept maps in Computer Engineering education to promote meaningful learning, creativity and collaboration*. Paper presented at the Frontiers in Education Conference (FIE), 2011.
- Canas, A. J., Carvalho, M., Arguedas, M., Leake, D. B., Maguitman, A., & Thomas, R. (2004). *Mining the web to suggest concepts during concept map construction*. Paper presented at the First International Conference on Concept Mapping, Pamplona, Spain.

- Canas, A. J., Ford, K. M., Novak, J. D., Hayes, P., Reichherzer, T., & Niranjana, S. (2001). Online Concept Maps: Enhancing collaborative learning by using technology with Concept Maps. *The Science Teacher*, 68(4), 49-51.
- Canas, A. J., Hill, G., Carff, R., Suri, N., Lott, J., & Eskridge, T. (2004). *CmapTools: A knowledge modeling and sharing environment*. Paper presented at the First International Conference on Concept Mapping, Pamplona, Spain.
- Canas, A. J., & Novak, J. D. (2006). *Re-examining the foundations for effective use of concept maps*. Paper presented at the Concept Maps: Theory, Methodology, Technology - Second International conference on Concept Mapping, San Jose, Costa Rica.
- Canas, A. J., & Novak, J. D. (2009a). Constructing your First Concept Map. 2011, from <http://cmap.ihmc.us/docs/ConstructingAConceptMap.html>
- Canas, A. J., & Novak, J. D. (2009b). Constructing your first concept map.
- Chang, K. E., Sung, Y.-T., & Chen, I.-D. (2002). The Effect of Concept Mapping to Enhance Text Comprehension and Summarization. *The Journal of Experimental Education*, 71(1), 5-23. doi: 10.1080/00220970209602054
- Chang, K. E., Sung, Y. T., & Chen, S. F. (2001). Learning through Computer-based Concept Mapping with Scaffolding Aid. *Journal of Computer Assisted Learning*, 17(1), 21-33.
- Chen, N.-S., Kinshuk, Wei, C.-W., & Chen, H.-J. (2008). Mining e-Learning domain concept map from academic articles. *Computers & Education*, 50(3), 1009-1021. doi: <http://dx.doi.org/10.1016/j.compedu.2006.10.001>
- Chen, S.-M., & Bai, S.-M. (2010). Using data mining techniques to automatically construct concept maps for adaptive learning systems. *Expert Systems with Applications*, 37(6), 4496-4503. doi: <http://dx.doi.org/10.1016/j.eswa.2009.12.060>
- Chung, G. K. W. K., O'Neil, H. F., & Herl, H. E. (1999). The Use of computer-based collaborative knowledge mapping to measure team processes and team outcomes. *Computers in Human Behavior*, 15(3), 463-493. doi: 10.1016/S0747-5632(99)00032-1
- Cimiano, P. (2006). *Ontology Learning and Population from Text: Algorithms, Evaluation and Applications*. New York: Springer Science and Business media, LLC.
- Cimiano, P., & Volker, J. (2005). *Text2Onto: a framework for ontology learning and data-driven change discovery*. Paper presented at the Proceedings of the 10th international conference on Natural Language Processing and Information Systems, Alicante, Spain.
- Clariana, R., & Koul, R. (2004). *A Computer-based approach for translating text into concept map-like representations*. Paper presented at the Proceedings of the First International Conference on Concept Mapping.
- Cliburn, J. W. (1990). Concept maps to promote meaningful learning. *Journal of College Science Teaching*, 19(6), 212-217.
- Coffey, J. W., Carnot, M. J., Feltovich, P. J., Feltovich, J., Hoffman, R. R., Cañas, A. J., & Novak, J. D. (2003). A Summary of Literature Pertaining to the Use of Concept Mapping Techniques and Technologies for Education and Performance Support. Chief of Naval Education and Training, Pensacola, FL.
- Coleman, E. B. (1998). Using Explanatory Knowledge During Collaborative Problem Solving in Science. *Journal of the Learning Sciences*, 7(3-4), 387-427. doi: 10.1080/10508406.1998.9672059
- Corbett, A., Trask, H., Scarpinato, K. C., & Hadley, W. (1998). A Formative Evaluation of the PACT Algebra II Tutor: Support for Simple Hierarchical Reasoning. In B. Goettl, H. Half, C. Redfield & V. Shute (Eds.), *Intelligent Tutoring Systems* (Vol. 1452, pp. 374-383): Springer Berlin Heidelberg.
- Culotta, A., & Sorensen, J. (2004). *Dependency tree kernels for relation extraction*. Paper presented at the Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics, Barcelona, Spain.
- Cunningham, H., Maynard, D., Bontcheva, K., & Tablan, V. (2002). *GATE: A framework and graphical development environment for robust NLP tools and applications*. Paper presented at the Proceedings of the 40th Annual Meeting of the ACL.

- Dabbagh, N. (2003). Scaffolding: An important teacher competency in online learning. *TechTrends*, 47(2), 39-44. doi: 10.1007/BF02763424
- Dali, L., & Fortuna, B. (2008). Triplet extraction from sentences using svm. *Proceedings of SiKDD*.
- Dali, L., Rusu, D., Fortuna, B., Mladenic, D., & Grobelnik, M. (2009). *Question Answering based on Semantic Graphs*. Paper presented at the Language and Technology Conference, Poznan, Poland.
- Dali, L., Rusu, D., Fortuna, B., Mladenic, D., & Grobelnik, M. (2010). AnswerArt - Contextualized Question Answering. In J. Balcázar, F. Bonchi, A. Gionis & M. Sebag (Eds.), *Machine Learning and Knowledge Discovery in Databases* (Vol. 6323, pp. 579-582): Springer Berlin Heidelberg.
- Davis, E. A. (1996). *Metacognitive Scaffolding To Foster Scientific Explanations [microform] / Elizabeth A. Davis*. [Washington, D.C.]: Distributed by ERIC Clearinghouse.
- Davis, E. A. (2000). Scaffolding students' knowledge integration: prompts for reflection in KIE. *International Journal of Science Education*, 22(8), 819-837. doi: 10.1080/095006900412293
- Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6), 391-407. doi: 10.1002/(SICI)1097-4571(199009)41:6<391::AID-ASII>3.0.CO;2-9
- Dennen, V. P. (2004). Cognitive Apprenticeship in Educational Practice: Research on Scaffolding, Modeling, Mentoring, and Coaching as Instructional Strategies *Handbook of Research on Educational Communications and Technology (2nd ed.)* (pp. 813-828). Mahwah, NJ, US: Lawrence Erlbaum Associates Publishers.
- Dias, P., & Sousa, A. P. (1997). Understanding navigation and disorientation in hypermedia learning environments. *J. Educ. Multimedia Hypermedia*, 6(2), 173-185.
- Diestel, R. (2005). *Graph Theory* (3 ed.): Springer.
- Dillon, J. T. (1988). *Questioning and Teaching: A manual of practice*. New York: Teachers College Press.
- Dunning, T. (1993). Accurate methods for the statistics of surprise and coincidence. *Comput. Linguist.*, 19(1), 61-74.
- Ebbinghaus, H. (1913). *Memory: A contribution to Experimental Psychology* (H. A. Ruger & C. E. Bussenius, Trans.): New York by Teachers College, Columbia University.
- Edmondson, K. M. (2005). Assessing science understanding through concept maps. In J. J. Mintzes, J. H. Wandersee & J. D. Novak (Eds.), *Assessing Science Understanding* (pp. 15-40). Burlington: Academic Press.
- Edmunds, A., & Morris, A. (2000). The problem of information overload in business organisations: a review of the literature. *International Journal of Information Management*, 20(1), 17-28. doi: [http://dx.doi.org/10.1016/S0268-4012\(99\)00051-1](http://dx.doi.org/10.1016/S0268-4012(99)00051-1)
- Etzioni, O., Cafarella, M., Downey, D., Popescu, A.-M., Shaked, T., Soderland, S., . . . Yates, A. (2005). Unsupervised named-entity extraction from the web: an experimental study. *Artif. Intell.*, 165(1), 91-134. doi: 10.1016/j.artint.2005.03.001
- Evens, M. W., Chang, R.-C., Lee, Y. H., Shim, L. S., Woo, C. W., Zhang, Y., . . . Rovick, A. A. (1997). *CIRCSIM-Tutor: an intelligent tutoring system using natural language dialogue*. Paper presented at the Proceedings of the fifth conference on Applied natural language processing: Descriptions of system demonstrations and videos, Washington, DC.
- Eylon, B.-S., & Reif, F. (1984). Effects of Knowledge Organization on Task Performance. *Cognition and Instruction*, 1(1), 5-44. doi: 10.1207/s1532690xci0101_2
- Ferry, B., Hedberg, J., & Harper, B. (1997). *How do preservice teachers use concept maps to organise their curriculum content knowledge*. Paper presented at the Australian Society for Computers in Learning in Tertiary Education, Perth, Australia.

- Fisher, K. M., Faletti, J., Patterson, H., Thornton, R., Lipson, J., & Spring, C. (1990). Computer-based concept mapping. *Journal of College Science Teaching*, 19(6), 347-352.
- Fisher, K. M., Wandersee, J. H., & Moody, D. E. (2002). *Mapping Biology Knowledge*: Springer Netherlands.
- Flavell, J. H., Miller, P. H., & Miller, S. A. (2002). *Cognitive Development*: Prentice Hall.
- Frantzi, K., Ananiadou, S., & Mima, H. (2000). Automatic recognition of multi-word terms: the C-value/NC-value method. *International Journal on Digital Libraries*, 3(2), 115-130. doi: 10.1007/s007999900023
- Frey, B. A., & Birnbaum, D. J. (2002). Learners' perceptions on the value of PowerPoint in Lectures.
- Fundel, K., Kuffer, R., Zimmer, R., & Miyano, S. (2007). RelEx---Relation extraction using dependency parse trees. *Bioinformatics*, 23(3), 365-371. doi: 10.1093/bioinformatics/btl616
- Gallagher, B. (2006). Matching structure and semantics: A survey on graph-based pattern matching. *AAAI FS*, 6, 45-53.
- Gantayat, N., & Iyer, S. (2011, 14-16 July 2011). *Automated Building of Domain Ontologies from Lecture Notes in Courseware*. Paper presented at the Technology for Education (T4E), 2011 IEEE International Conference on.
- Ge, N., Hale, J., & Charniak, E. (1998). *A statistical approach to anaphora resolution*. Paper presented at the Sixth workshop on very large corpora.
- Ge, X., & Land, S. (2003). Scaffolding students' problem-solving processes in an ill-structured task using question prompts and peer interactions. *Educational Technology Research and Development*, 51(1), 21-38. doi: 10.1007/BF02504515
- Graesser, A. C., Person, N., & Huber, J. (1992). Mechanisms that generate questions. In T. W. Lauer, E. Peacock & A. C. Graesser (Eds.), *Questions and information systems* (pp. 167-187). Hillsdale, NJ, England: Lawrence Erlbaum Associates, Inc.
- Hall, R. H., & O'Donnell, A. M. (1996). Cognitive and Affective Outcomes of Learning from Knowledge Maps. *Contemporary Educational Psychology*, 21(1), 94-101. doi: <http://dx.doi.org/10.1006/ceps.1996.0008>
- Hannafin, M., Land, S., & Oliver, K. (1999). Open learning environments: Foundations, methods, and models. In M. R. Charles (Ed.), *Instructional-design theories and models: A new paradigm of instructional theory* (Vol. 2, pp. 115-140). Mahwah, NJ: Lawrence Erlbaum.
- Hauser, S., Nuckles, M., & Renkl, A. (2006). *Supporting concept mapping for learning from text*. Paper presented at the Proceedings of the 7th international conference on Learning sciences, Bloomington, Indiana.
- Hearst, M. A. (1992). *Automatic acquisition of hyponyms from large text corpora*. Paper presented at the Proceedings of the 14th conference on Computational linguistics - Volume 2, Nantes, France.
- Hearst, M. A. (2000). The Debate on Automated Essay Grading. *IEEE Intelligent Systems and their Applications*, 15(5), 22-37. doi: 10.1109/5254.889104
- Heinze-Fry, J. A., & Novak, J. D. (1990). Concept mapping brings long-term movement toward meaningful learning. *Science Education*, 74(4), 461-472. doi: 10.1002/sce.3730740406
- Herl, H. E., O'Neil, H. F., Chung, G. K. W. K., Dennis, R. A., & Lee, J. J. (1997). Feasibility of an On-line Concept Mapping Construction and Scoring System. Annual Meeting of the American Educational Research Association, Chicago, IL.
- Hogan, K., & Pressley, M. (1997). Scaffolding scientific competencies within classroom communities of inquiry. In K. Hogan & M. Pressley (Eds.), *Scaffolding student learning: Instructional approaches and issues* (pp. 74-107). Cambridge, MA: Brookline Books.
- Horton, P. B., McConney, A. A., Gallo, M., Woods, A. L., Senn, G. J., & Hamelin, D. (1993). An investigation of the effectiveness of concept mapping as an instructional tool. *Science Education*, 77(1), 95-111. doi: 10.1002/sce.3730770107

- Hripcsak, G., & Rothschild, A. (2005). Agreement, the f-measure, and reliability in information retrieval. *J. of Am. Medical Informatics Association*, 12(3), 296-298. doi: citeulike-article-id:4491320
- Hsu, L.-L. (2004). Developing concept maps from problem-based learning scenario discussions. *Journal of Advanced Nursing*, 48(5), 510-518.
- Huang, H.-W., Wu, C.-W., & Chen, N.-S. (2012). The effectiveness of using procedural scaffoldings in a paper-plus-smartphone collaborative learning context. *Comput. Educ.*, 59(2), 250-259. doi: 10.1016/j.compedu.2012.01.015
- Hunkins, F. P. (1972). *Questioning strategies and techniques*. Boston, MA: Allyn and Bacon.
- Ide, N., & Veronis, J. (1998). Introduction to the special issue on word sense disambiguation: the state of the art. *Computational Linguistics*, 24(1), 2-40.
- Irvine, L. M. C. (1995). Can concept mapping be used to promote meaningful learning in nurse education? *Journal of Advanced Nursing*, 21(6), 1175-1179. doi: 10.1046/j.1365-2648.1995.21061175.x
- Jumaat, N. F., & Tasir, Z. (2014, 11-13 April 2014). *Instructional Scaffolding in Online Learning Environment: A Meta-analysis*. Paper presented at the Teaching and Learning in Computing and Engineering (LaTiCE), 2014 International Conference on.
- Jurafsky, D., & Martin, J., H. (2009). *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition* (2nd Edition ed.): Pearson Education International.
- Kambhatla, N. (2004). *Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations*. Paper presented at the Proceedings of the ACL 2004 on Interactive poster and demonstration sessions, Barcelona, Spain.
- Kasinathan, V., Mustapha, A., & Rani, M. F. (2013). Structure-based Algorithm for Presentation Mapping in Graphical Knowledge Display. *International Journal of Information and Education Technology*, 3(2), 196-200. doi: 10.7763/IJET.2013.V3.263
- Katz, B. (1997). *Annotating the World Wide Web using Natural Language*. Paper presented at the Fifth RIAO Conference on Computer Assisted Information Searching on the Internet.
- Kenny, D. A. (1987). *Statistics for the Social and Behavioural Sciences*.
- Kinchin, I. M. (2001). If concept mapping is so helpful to learning biology, why aren't we all doing it? *International Journal of Science Education*, 23(12), 1257-1269. doi: 10.1080/09500690010025058
- Kinchin, I. M. (2006a). Concept mapping, PowerPoint, and a pedagogy of access. *Journal of Biological Education*, 40(2), 79-83. doi: 10.1080/00219266.2006.9656018
- Kinchin, I. M. (2006b). Developing PowerPoint handouts to support meaningful learning. *British Journal of Educational Technology*, 37(4), 647-650. doi: 10.1111/j.1467-8535.2006.00536.x
- Kinchin, I. M., Chadha, D., & Kokotailo, P. (2008). Using PowerPoint as a lens to focus on linearity in teaching. *Journal of Further and Higher Education*, 32(4), 333-346. doi: 10.1080/03098770802392923
- Kinchin, I. M., Hay, D. B., & Adams, A. (2000). How a qualitative approach to concept map analysis can be used to aid learning by illustrating patterns of conceptual development. *Educational Research*, 42(1), 43-57. doi: 10.1080/001318800363908
- Kintsch, W., & Keenan, J. M. (1974). Recall of propositions as a function of their position in the hierarchical structure. *The representation of meaning in memory*, 137-140.
- Klein, D., & Manning, C. (2003a). *Accurate unlexicalized parsing*. Paper presented at the Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1, Sapporo, Japan.
- Klein, D., & Manning, C. (2003b). *Fast Exact Inference with a Factored Model for Natural Language Parsing*. Paper presented at the Advances in Neural Information Processing Systems.

- Kurose, J. F., & Ross, K. W. (2012). *Computer Network: A Top-Down Approach* (6th edition ed.): Addison-Wesley.
- Lambiotte, J. G., & Dansereau, D. F. (1992). Effects of Knowledge Maps and Prior Knowledge on Recall of Science Lecture Content. *The Journal of Experimental Education*, 60(3), 189-201. doi: 10.1080/00220973.1992.9943875
- Larraza-Mendiluze, E., & Garay-Vitoria, N. (2013). *Use of concept maps to analyze students' understanding of the I/O subsystem*. Paper presented at the Proceedings of the 13th Koli Calling International Conference on Computing Education Research, Koli, Finland.
- Lassila, O., & Swick, R. R. (1999). Resource Description Framework (RDF) Model and Syntax Specification 2014
- Lau, R. Y. K., Chung, A. Y. K., Song, D., & Huang, Q. (2008). Towards Fuzzy Domain Ontology Based Concept Map Generation for E-Learning. In H. Leung, F. Li, R. Lau & Q. Li (Eds.), *Advances in Web Based Learning – ICWL 2007* (Vol. 4823, pp. 90-101): Springer Berlin Heidelberg.
- Leake, D. B., Gabriela, M. A., & Thomas, R. (2004). *Understanding Knowledge Models: Modeling Assessment of Concept Importance in Concept Maps*. Paper presented at the Twenty-Sixth Annual Conference of the Cognitive Science Society.
- Leake, D. B., Maguitman, A., Reichherzer, T., Canas, A. J., Carvalho, M., Arguedas, M., . . . Eskridge, T. (2003). *Aiding knowledge capture by searching for extensions of knowledge models*. Paper presented at the Proceedings of the 2nd international conference on Knowledge capture, Sanibel Island, FL, USA.
- Lee, C.-H., Lee, G.-G., & Leu, Y. (2009). Application of automatically constructed concept map of learning to conceptual diagnosis of e-learning. *Expert Systems with Applications*, 36(2, Part 1), 1675-1684. doi: <http://dx.doi.org/10.1016/j.eswa.2007.11.049>
- Leelawong, K., & Biswas, G. (2008). Designing Learning by Teaching Agents: The Betty's Brain System. *International Journal of Artificial Intelligence in Education*, 18(3), 181-208.
- Lepper, M. R., Drake, M. F., & O'Donnell-Johnson, T. (1997). Scaffolding techniques of expert human tutors. In K. Hogan & M. Pressley (Eds.), *Scaffolding student learning: Instructional approaches and issues* (pp. 108-144). Cambridge, MA, US: Brookline Books.
- Leskovec, J., Grobelnik, M., & Milic-Frayling, N. (2004). *Learning Sub Structures of Document Semantic Graphs for Document Summarization*. Paper presented at the KDD 2004 workshop on Link Analysis and Group Detection (LinkKDD), Seattle, WA, USA.
- Lin, D. (2003). Dependency-Based Evaluation of Minipar. In A. Abeillé (Ed.), *Treebanks* (Vol. 20, pp. 317-329): Springer Netherlands.
- Lopez, V., Uren, V., Motta, E., & Pasin, M. (2007). AquaLog: An ontology-driven question answering system for organizational semantic intranets. *Web Semantics: Science, Services and Agents on the World Wide Web*, 5(2), 72-105. doi: <http://dx.doi.org/10.1016/j.websem.2007.03.003>
- Luchini, K., Quintana, C., Krajcik, J., Farah, C., Nandihalli, N., Reese, K., . . . Soloway, E. (2002). *Scaffolding in the small: designing educational supports for concept mapping on handheld computers*. Paper presented at the CHI '02 Extended Abstracts on Human Factors in Computing Systems, Minneapolis, Minnesota, USA.
- Macnamara, J. (1982). *Names for things: A study of human learning*. Cambridge, MA: M.I.T. Press.
- Manning, C., Raghavan, P., & Schütze, H. (2008). *Introduction to Information Retrieval*: Cambridge University Press.
- Manning, C., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., & McClosky, D. (2014). *The Stanford CoreNLP Natural Language Processing toolkit*. Paper presented at the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, Baltimore, Maryland.

- Marcus, M., Kim, G., Marcinkiewicz, M. A., MacIntyre, R., Bies, A., Ferguson, M., . . . Schasberger, B. (1994). *The Penn Treebank: annotating predicate argument structure*. Paper presented at the Proceedings of the workshop on Human Language Technology, Plainsboro, NJ.
- Markham, K. M., Mintzes, J. J., & Jones, M. G. (1994). The concept map as a research and evaluation tool: Further evidence of validity. *Journal of Research in Science Teaching*, *31*(1), 91-101. doi: 10.1002/tea.3660310109
- Martin, J., Slemon, A. G., Hiebert, B., Hallberg, E. T., & Cummings, A. L. (1989). Conceptualizations of novice and experienced counselors. *Counseling Psychology*, *36*, 395-400.
- Martin, P. A. (2008). *Use of Semantic Networks as Learning Material and Evaluation of the Approach by Students*. Paper presented at the International Conference on Open Learning and Distance Education, Vienna, Austria.
- Maynard, D., Li, Y., & Peters, W. (2008). *NLP Techniques for Term Extraction and Ontology Population*. Paper presented at the Proceedings of the 2008 conference on Ontology Learning and Population: Bridging the Gap between Text and Knowledge.
- McClure, J. R., Sonak, B., & Suen, H. K. (1999). Concept map assessment of classroom learning: Reliability, validity, and logistical practicality. *Journal of Research in Science Teaching*, *36*(4), 475-492. doi: 10.1002/(SICI)1098-2736(199904)36:4<475::AID-TEA5>3.0.CO;2-O
- Meyer, B. J. F., Brandt, D. M., & Bluth, G. J. (1980). Use of top level structure in text: Key for reading comprehension in ninth grade students. *Reading Research Quarterly*, *16*, 72-103.
- Miller, G. A., Beckwith, R., Fellbaum, C., Gross, D., & Miller, K. J. (1990). Introduction to WordNet: An On-line Lexical Database*. *International Journal of Lexicography*, *3*(4), 235-244. doi: 10.1093/ijl/3.4.235
- Mitkov, R. (1998). *Robust pronoun resolution with limited knowledge*. Paper presented at the Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 2, Montreal, Quebec, Canada.
- Mitrović, A. (1998). Experiences in Implementing Constraint-Based Modeling in SQL-Tutor. In B. Goettl, H. Half, C. Redfield & V. Shute (Eds.), *Intelligent Tutoring Systems* (Vol. 1452, pp. 414-423): Springer Berlin Heidelberg.
- Mitrovic, A., Koedinger, K., & Martin, B. (2003). A Comparative Analysis of Cognitive Tutoring and Constraint-Based Modeling. In P. Brusilovsky, A. Corbett & F. de Rosi (Eds.), *User Modeling 2003* (Vol. 2702, pp. 313-322): Springer Berlin Heidelberg.
- Moodle. (2014). Quiz module. 2014, from https://docs.moodle.org/27/en/Quiz_module
- Naveh-Benjamin, M., Lin, Y., & McKeachie, W. J. (1995). Inferring Students' Cognitive Structures and their Development using the 'Fill in the Structure' (F.I.T.S) technique. In P. D. Nichols, S. F. Chipman & R. L. Brennan (Eds.), *Cognitively diagnostic assessment* (pp. 279-304). Hillsdale, NJ: Lawrence Erlbaum Associates.
- Nesbit, J. C., & Adesope, O. (2006). Learning with Concept and Knowledge Maps: A Meta-Analysis. *Review of Educational Research*, *76*(3), 413-448.
- Novak, J. D. (1990). Concept mapping: A useful tool for science education. *Journal of Research in Science Teaching*, *27*(10), 937-949. doi: 10.1002/tea.3660271003
- Novak, J. D. (1993). Human constructivism: A unification of psychological and epistemological phenomena in meaning making. *International Journal of Personal Construct Psychology*, *6*(2), 167-193. doi: 10.1080/08936039308404338
- Novak, J. D. (1998). *Learning, Creating and Using Knowledge: Concept maps as Facilitative tools in Schools and Corporations*. Mahwah, NJ: Lawrence Erlbaum Associates.
- Novak, J. D. (2002). Meaningful learning: The essential factor for conceptual change in limited or inappropriate propositional hierarchies leading to empowerment of learners. *Science Education*, *86*(4), 548-571. doi: 10.1002/sce.10032

- Novak, J. D. (2010). *Learning, Creating, and Using Knowledge: Concept Maps as Facilitative Tools in Schools and Corporations* (2 ed.): NY: Routledge.
- Novak, J. D., & Canas, A. J. (2006). The Theory underlying Concept maps and How to construct and use them. Pensacola FL: Florida Institute for Human and Machine Cognition.
- Novak, J. D., & Gowin, D. B. (1984). *Learning How to Learn*: Cambridge and NY: Cambridge University Press.
- O'Donnell, A. M., & Dansereau, D. F. (2000). Interactive Effects of Prior Knowledge and Material Format on Cooperative Teaching. *The Journal of Experimental Education*, 68(2), 101-118. doi: 10.2307/20152621
- O'Donnell, A. M., Dansereau, D. F., & Hall, R. H. (2002). Knowledge Maps as Scaffolds for Cognitive Processing. *Educational Psychology Review*, 14(1), 71-86. doi: 10.1023/A:1013132527007
- Okebukola, P. A. (1992). Can Good Concept Mappers be Good Problem Solvers in Science? *Educational Psychology*, 12(2), 113-129. doi: 10.1080/0144341920120203
- Olney, A. M. (2010). *Extraction of concept maps from textbooks for domain modeling*. Paper presented at the Proceedings of the 10th international conference on Intelligent Tutoring Systems - Volume Part II, Pittsburgh, PA.
- Olney, A. M., C., G. A., & K., P. N. (2012). Question Generation from Concept Maps. *Dialogue and Discourse*, 3(2), 79-99.
- Olney, A. M., Cade, W. L., & Williams, C. (2011). *Generating concept map exercises from textbooks*. Paper presented at the Proceedings of the 6th Workshop on Innovative Use of NLP for Building Educational Applications, Portland, Oregon.
- Ono, M., Harada, F., & Shimakawa, H. (2011). Semantic Network to Formalize Learning Items from Lecture Notes. *Advanced Computer Science*, 1(1), 10-15.
- Ormrod, J. E. (2011). *Human Learning*. New Jersey, USA: Pearson Education Inc.
- Pankratius, W. J. (1990). Building an organized knowledge base: Concept mapping and achievement in secondary school physics. *Journal of Research in Science Teaching*, 27(4), 315-333. doi: 10.1002/tea.3660270404
- Park, U., & Calvo, R. A. (2008). *Automatic Concept Map Scoring Framework Using the Semantic Web Technologies*. Paper presented at the Proceedings of the 2008 Eighth IEEE International Conference on Advanced Learning Technologies.
- Pass, F. G. (1992). Training strategies for attaining transfer of problem-solving skill in statistics: A cognitive-load approach. *Educational Psychology*, 84(4), 429-434. doi: 10.1037/0022-0663.84.4.429
- Patterson, M. E., Dansereau, D. F., & Wiegmann, D. A. (1993). Receiving information during a cooperative episode: Effects of communication aids and verbal ability. *Learning and Individual Differences*, 5(1), 1-11. doi: [http://dx.doi.org/10.1016/1041-6080\(93\)90022-K](http://dx.doi.org/10.1016/1041-6080(93)90022-K)
- Pearsall, N. R., Skipper, J. E. J., & Mintzes, J. J. (1997). Knowledge restructuring in the life sciences: A longitudinal study of conceptual change in biology. *Science Education*, 81(2), 193-215. doi: 10.1002/(SICI)1098-237X(199704)81:2<193::AID-SCE5>3.0.CO;2-A
- Person, N., Olney, A. M., D'Mello, S., & Lehman, B. (2012). *Interactive Concept Maps and Learning Outcomes in Guru*. Paper presented at the Twenty-fifth International FLAIRS Conference Menlo Park, CA.
- Porter, M. F. (2006). An algorithm for suffix stripping. *Program: Electronic Library and Information Systems*, 40(3), 211-218.
- Quintana, C., Reiser, B. J., Davis, E. A., Krajcik, J., Fretz, E., Duncan, R. G., . . . Soloway, E. (2004). A Scaffolding Design Framework for Software to Support Science Inquiry. *Journal of the Learning Sciences*, 13(3), 337-386. doi: 10.1207/s15327809jls1303_4
- Regoczei, S. B., & Hirst, G. (1988). *Knowledge Acquisition as Knowledge Explication by Conceptual Analysis*. University of Toronto: Computer Systems Research Institute.

- Regoczei, S. B., & Hirst, G. (1992). Knowledge and Knowledge Acquisition in the Computational Context. In R. Hoffman (Ed.), *The Psychology of Expertise* (pp. 12-25): Springer New York.
- Riloff, E., & Shepherd, J. (1997). *A corpus-based approach for building semantic lexicons*. Paper presented at the Empirical methods in Natural Language Processing (EMNLP).
- Rosenfeld, B., Feldman, R., & Aumann, Y. (2002). *Structural extraction from visual layout of documents*. Paper presented at the Proceedings of the eleventh international conference on Information and knowledge management, McLean, Virginia, USA.
- Ross, B., & Munby, H. (1991). Concept mapping and misconceptions: a study of high-school students' understandings of acids and bases. *International Journal of Science Education*, 13(1), 11-23. doi: 10.1080/0950069910130102
- Roth, W.-M., & Roychoudhury, A. (1993). The concept map as a tool for the collaborative construction of knowledge: A microanalysis of high school physics students. *Journal of Research in Science Teaching*, 30(5), 503-534. doi: 10.1002/tea.3660300508
- Ruiz-Primo, M. A. (2004). *Examining concept maps as an assessment tool*. Paper presented at the Concept Maps: Theory, Methodology, Technology; First International Conference on Concept Mapping, Pamplona, Spain.
- Ruiz-Primo, M. A., Schultz, S. E., Li, M., & Shavelson, R. J. (2001). Comparison of the reliability and validity of scores from two concept-mapping techniques. *Research in Science Teaching*, 38(2), 260-278.
- Ruiz-Primo, M. A., & Shavelson, R. J. (1996). Problems and issues in the use of concept maps in science assessment. *Journal of Research in Science Teaching*, 33(6), 569-600. doi: 10.1002/(SICI)1098-2736(199608)33:6<569::AID-TEA1>3.0.CO;2-M
- Rusu, D., Dali, L., Fortuna, B., Marko, G., & Dunja, M. (2007). *Triplet Extraction from Sentences*. Paper presented at the Data mining and Data warehouses, Ljubljana, Slovenia.
- Salomon, G., Globerson, T., & Guterman, E. (1989). The computer as a zone of proximal development: Internalizing reading-related metacognitions from a reading partner. *Journal of Education Psychology*, 81(4), 620-627.
- Salton, G., & Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *International Journal of Information Processing and Management*, 24(5), 513-523. doi: 10.1016/0306-4573(88)90021-0
- Salton, G., & McGill, M. J. (1986). *Introduction to Modern Information Retrieval*: McGraw-Hill, Inc.
- Sanders, K., Boustedt, J., Eckerdal, A., McCartney, R., Mostr, J. E., Thomas, L., & Zander, C. (2008). Student understanding of object-oriented programming as expressed in concept maps. *SIGCSE Bull.*, 40(1), 332-336. doi: 10.1145/1352322.1352251
- Sanderson, M., & Croft, B. (1999). *Deriving concept hierarchies from text*. Paper presented at the Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, Berkeley, California, USA.
- Sang, E. F. T. K., & Meulder, F. D. (2003). *Introduction to the CoNLL-2003 shared task: language-independent named entity recognition*. Paper presented at the Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003 - Volume 4, Edmonton, Canada.
- Sawyer, R. K. (2006). *The Cambridge Handbook of the Learning Sciences*. New York: Cambridge University Press.
- Schacter, D. L., Gilbert, D. T., & Wagner, D. M. (2009). *Psychology*. New York: Worth.
- Schvaneveldt, R. W. E. (Ed.). (1990). *Pathfinder associative networks: studies in knowledge organization*: Ablex Publishing Corp.
- Silberchatz, A., Galvin, P. B., & Gagne, G. (2012). *Operating System Concepts* (9th Edition ed.): John Wiley and Sons, Inc.
- Sleator, D., & Temperley, D. (1991). *Parsing English with a Link Grammar*: Carnegie Mellon University.

- Snow, R., Jurafsky, D., & Ng, A. (2005). Learning Syntactic Patterns for Automatic Hypernym Discovery. In L. Saul, Y. Weiss & L. Bottou (Eds.), *Advances in Neural Information Processing Systems 17* (pp. 1297-1304): MIT Press.
- Sommerville, I. (2010). *Software Engineering* (9th Edition ed.): Addison-Wesley.
- Stalling, W. (2007). *Data and Computer Communications* (6th edition ed.). Upper Saddle River, NJ: Prentice Hall.
- Starr, R. R., & Parente de Oliveira, J. M. (2013). Concept maps as the first step in an ontology construction method. *Information Systems*, 38(5), 771-783. doi: <http://dx.doi.org/10.1016/j.is.2012.05.010>
- Sugahara, S., & Boland, G. (2006). The Effectiveness of PowerPoint presentations in the Accounting Classroom. *Accounting Education*, 15(4), 391-403. doi: 10.1080/09639280601011099
- Taber, K., S. . (1994). Student reaction on being introduced to concept mapping. *Physics Education*, 29(5), 276.
- Tan, P.-N., Steinbach, M., & Kumar, V. (2005). *Introduction to Data Mining*: Addison-Wesley.
- Tchounikine, P. (2011). *Computer Science and Educational Software Design: A resource for multidisciplinary work in technology enhanced learning*: Springer-Verlag.
- Toutanova, K., Klein, D., Manning, C., & Singer, Y. (2003). *Feature-rich part-of-speech tagging with a cyclic dependency network*. Paper presented at the Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology Edmonton, Canada.
- Tseng, S.-S., Sue, P.-C., Su, J.-M., Weng, J.-F., & Tsai, W.-N. (2007). A new approach for constructing the concept map. *Computers & Education*, 49(3), 691-707. doi: <http://dx.doi.org/10.1016/j.compedu.2005.11.020>
- Tseng, Y.-H., Chang, C.-Y., Rundgren, S.-N. C., & Rundgren, C.-J. (2010). Mining concept maps from news stories for measuring civic scientific literacy in media. *Comput. Educ.*, 55(1), 165-177. doi: 10.1016/j.compedu.2010.01.002
- Tufte, E. R. (2003). *The Cognitive Style of PowerPoint*: Graphics Press.
- Ullmann, J. R. (1976). An Algorithm for Subgraph Isomorphism. *J. ACM*, 23(1), 31-42. doi: 10.1145/321921.321925
- Valerio, A., & Leake, D. B. (2006). *Jump-starting Concept Map Construction with Knowledge Extracted from Documents*. Paper presented at the Proceedings of the Second International Conference on Concept Mapping.
- Valerio, A., & Leake, D. B. (2012). *Using Automatically Generated Concept Maps for Document Understanding: A Human Subjects Experiment*. Paper presented at the Fifth International Conference on Concept Mapping, Valletta, Malta.
- Villalon, J. (2011). Question types: Concept Map Question types. 2012, from https://moodle.org/plugins/view.php?plugin=qtype_conceptmap
- Villalon, J., & Calvo, R. A. (2008, 9-12 Dec. 2008). *Concept Map Mining: A Definition and a Framework for Its Evaluation*. Paper presented at the Web Intelligence and Intelligent Agent Technology, 2008. WI-IAT '08. IEEE/WIC/ACM International Conference on.
- Villalon, J., & Calvo, R. A. (2009). *Concept Extraction from Student Essays, Towards Concept Map Mining*. Paper presented at the Proceedings of the 2009 Ninth IEEE International Conference on Advanced Learning Technologies.
- Villalon, J., & Calvo, R. A. (2011). Concept maps as cognitive visualisations of writing assignments. *Educational technology and Society*, 14(3), 16-27.
- Villalon, J., Calvo, R. A., & Montenegro, R. (2010). *Analysis of a gold standard for concept map mining—How humans summarize text using concept maps*. Paper presented at the Fourth International Conference on Concept Mapping (CMC'10), Viña del Mar, Chile.
- Vygotsky, L. S. (1978). *Mind in society: The development of higher psychological process*. Cambridge, MA: Harvard University Press.
- Wang, W. M., Cheung, C. F., Lee, W. B., & Kwok, S. K. (2008). Mining knowledge from natural language texts using fuzzy associated concept mapping. *Information Processing & Management*, 44(5), 1707-1719. doi: <http://dx.doi.org/10.1016/j.ipm.2008.05.002>

- Weinbrenner, S., Engler, J., & Hoppe, H. U. (2011). Ontology-Supported Scaffolding of Concept Maps. In G. Biswas, S. Bull, J. Kay & A. Mitrovic (Eds.), *Artificial Intelligence in Education* (Vol. 6738, pp. 582-584): Springer Berlin Heidelberg.
- Wilen, W. W. (1986). *Questioning skills for teachers* (2nd ed. ed.). WASHINGTON, D. C.: National Education Association.
- Wolf, C. (2003). *iWeaver: towards 'learning style'-based e-learning in computer science education*. Paper presented at the Proceedings of the fifth Australasian conference on Computing education - Volume 20, Adelaide, Australia.
- Wood, D., Bruner, J. S., & Ross, G. (1976). THE ROLE OF TUTORING IN PROBLEM SOLVING*. *Journal of Child Psychology and Psychiatry*, 17(2), 89-100. doi: 10.1111/j.1469-7610.1976.tb00381.x
- Yoo, J. S., & Cho, M.-H. (2012). *Mining Concept Maps to Understand University Students' Learning*. Paper presented at the 5th International Conference on Educational Data Mining, Chania, Greece.
- Zelenko, D., Aone, C., & Richardella, A. (2002). *Kernel methods for relation extraction*. Paper presented at the Proceedings of the ACL-02 conference on Empirical methods in natural language processing - Volume 10.
- Zhang, M., & Quintana, C. (2012). Scaffolding strategies for supporting middle school students' online inquiry processes. *Computers & Education*, 58(1), 181-196. doi: <http://dx.doi.org/10.1016/j.compedu.2011.07.016>
- Zouaq, A., Gasevic, D., & Hatala, M. (2012). Voting Theory for Concept Detection. In E. Simperl, P. Cimiano, A. Polleres, O. Corcho & V. Presutti (Eds.), *The Semantic Web: Research and Applications* (Vol. 7295, pp. 315-329): Springer Berlin Heidelberg.
- Zouaq, A., & Nkambou, R. (2008). Building Domain Ontologies from Text for Educational Purposes. *Learning Technologies, IEEE Transactions on*, 1(1), 49-62. doi: 10.1109/TLT.2008.12
- Zouaq, A., & Nkambou, R. (2009). Evaluating the Generation of Domain Ontologies in the Knowledge Puzzle Project. *Knowledge and Data Engineering, IEEE Transactions on*, 21(11), 1559-1572. doi: 10.1109/TKDE.2009.25
- Zubrinic, K., Kalpic, D., & Milicevic, M. (2012). Review: The automatic creation of concept maps from documents written using morphologically rich languages. *Expert Syst. Appl.*, 39(16), 12709-12718. doi: 10.1016/j.eswa.2012.04.065