Enhancing Factoid Question Answering Using Frame Semantic-Based Approaches

Dissertation

Submitted at the Graduate School of Information Technology and Mathematical Sciences of University of Ballarat in fulfilment of the requirements for the degree of Doctor of Philosophy

Bahadorreza Ofoghi

September 2009

Statement of Originality

Except where explicit references are made, the text of this thesis contains no material published elsewhere or extracted in whole or in part from a thesis by which I have qualified for or have been awarded another degree or diploma. No other person's work has been relied upon or used without due acknowledgement in the main text and bibliography of the thesis.

Bahadorreza Ofoghi
September 2009

Source Material

Parts of the material presented in this thesis have been published in the following papers during my PhD candidature:

- Ofoghi, Bahadorreza, John Yearwood, and Liping Ma (2009). "The impact of frame semantic annotation, frame alignment techniques, and fusion methods on factoid answer processing." In: *Journal of the American Society for Information Science and Technology (JASIST)* 60.2, pp. 247-263.
- Ofoghi, Bahadorreza, John Yearwood, and Liping Ma (2008*b*). "The impact of semantic class identification and semantic role labeling on natural language answer extraction." In: *30th European Conference on Information Retrieval (ECIR 2008).* Glasgow, Scotland, pp. 430-437.
- Ofoghi, Bahadorreza, John Yearwood, and Liping Ma (2008*a*). "FrameNet-based fact-seeking answer processing: A study of semantic alignment techniques and lexical coverage." In: *21st Australasian Joint Conference on Artificial Intelligence (AI 2008).* Auckland, New Zealand, pp. 192-201.
- Ofoghi, Bahadorreza, John Yearwood, and Liping Ma (2007). "Two-step comprehensive open domain text annotation with frame semantics." In: *Australasian Language Technology Workshop 2007.* Melbourne, Australia, pp. 83-91.
- Ofoghi, Bahadorreza, John Yearwood, and Ranadhir Ghosh (2007). "A within-frame ontological extension on FrameNet: Application in predicate chain analysis and question answering." In: *20th Australian Joint Conference on Artificial Intelligence (AI 2007).* Griffith University, QLD, Australia, pp. 404-414.
- Ofoghi, Bahadorreza, John Yearwood, and Ranadhir Ghosh (2006*c*). "A semantic method to information extraction for decision support systems." In: *12th Americas Conference on Information Systems (AMCIS 2006).* Acapulco, Mexico, pp. 1475-1481.
- Ofoghi, Bahadorreza, John Yearwood, and Ranadhir Ghosh (2006*b*). "A semantic approach to boost passage retrieval effectiveness for question answering." In: *29th Australian Computer Science Conference.* Vol. 48. Hobart, Tasmania, Australia, pp. 95-101.

- Ofoghi, Bahadorreza, John Yearwood, and Ranadhir Ghosh (2006*a*). "A hybrid question answering schema using encapsulated semantics in lexical resources." In: *19th Australian Joint Conference on Artificial Intelligence (AI 2006)*. Vol. 4304/2006. Hobart, Tasmania, Australia, pp. 1276-1280.

Abstract

As the rapid rise of information stored in document databases continues, there is a real possibility of using these textual databases in systems that automatically provide answers to questions issued by users in natural language. Identification of candidate answers - within these document repositories - to natural language factoid questions using Question Answering (QA) systems is a challenging task that has been tackled by many researchers. One of the problems in this domain is to retrieve text passages that potentially contain answers to the questions. From an information retrieval viewpoint retrieval of such passages requires more comprehensive analysis than retrieving related passages based on surface syntactic structures of the texts. Another problem in factoid QA is the extraction of the text excerpts that are highly likely to answer factoid questions given the different syntactic and semantic structures that can be used in questions and passages.

These two problems have attracted significant attention in recent years, especially in the communities of natural language processing and computational linguistics. It is understood that the above-mentioned tasks can be more effectively handled by using tools, methods, and resources from the linguistics domains. Linguistic resources can bring human-like understanding of texts and useful world knowledge into the domain of QA to provide greater semantic capability in dealing with text-based challenges.

In this thesis, FrameNet is used to enhance the performance of semantic QA systems. FrameNet is a linguistic resource that encapsulates Frame Semantics and provides scenario-based generalizations over lexical items that share similar semantic backgrounds. By using the concepts and the elements of FrameNet (for query reformulation) we tackle the problem of answer passage retrieval in an effective way that shows an improvement over non-semantic state-of-the-art passage retrieval methods. This is performed after exploitation of different keyword-based, syntactic, and topical features in enhancing a well-established passage retrieval method (MultiText). We consider some other techniques, implemented in the Lemur toolkit, for comparison purposes.

We also exploit the FrameNet resource in identification, extraction, and scoring of text snippets from answer passages that are likely answer candidates to factoid questions. One of our new FrameNet-based answer processing techniques shows improvement over the performance of existing FrameNet-based methods. The underlying difficulty of semantic parsing is considered by investigating the effects of different levels of shallow semantic parsing (that can be achieved based on

FrameNet frames and frame elements) on the outcomes of this work. We also study the possible benefits of fusing FrameNet-based answer processing techniques with other non-semantic models of answer extraction and scoring. This work demonstrates that FrameNet-based shallow semantic approaches in combination with other approaches (such as Named Entity-based approaches) can deliver enhanced performance in factoid QA systems.

In terms of FrameNet development, we conduct some studies to observe the current shortcomings of FrameNet that interfere with FrameNet-based factoid QA performances. Lexical coverage of different part-of-speech predicates is analysed in different FrameNet versions (1.2 and 1.3). This shows that noun predicates require more attention in the future in order to take greater advantage of FrameNet for the task of factoid QA.

Dedicated to

my father Jahangir, my mother Nasrin, my wife Armita, and my daughter Rosha

Acknowledgements

I would like to specially thank my principal supervisor Professor John Yearwood who was the greatest supporter for me both in the scientific direction and all related non-scientific issues during my wonderful candidature at the University of Ballarat. John taught me how to conduct a research study in a field which has attracted many well-known experts without getting disappointed. I also learnt from him how to be patient and keep hope when experiencing the tough side of research. He also instructed me on producing high quality academic writings. I really appreciate all your guidance, support, and patience especially for my language as a non-native English speaker.

I thank my wonderful associate supervisors Dr. Ranadhir Ghosh and Dr. Liping Ma for their great ideas and assistance. After Ranadhir's superb assistance in the first year, Liping gave me her best thoughts and support during the following two years. She also helped me in presenting my work and writing research papers. I had the good fortune to work with Ranadhir and Liping along with my principal supervisor.

A very big thank you to Assistant Professor Katrin Erk from the University of Texas at Austin for always being there to kindly reply to my questions especially when I needed to consult with those who have been in the domain for a longer period of time. She also patiently provided me with a lot of guidance on the SHALMANESER shallow semantic parser and the SALTO annotation tool. Thanks to her previous colleague Sebastian Pado as well for his great effort in implementing SHALMANESER and helping Katrin in giving me solutions on using the parser.

It would have been hardly possible for me to study my PhD without receiving different scholarships from the University of Ballarat and Centre for Informatics and Applied Optimization in the Graduate School of Information Technology and Mathematical Sciences (GSITMS). Therefore, I really thank all the people who supported me and kindly offered the grants. Thanks to Professor Sidney Morris, Head of GSITMS, for his support and making it possible for me to attend different related events and meet other people working in the domain of my study.

I am also grateful to the people from the Berkley FrameNet project for providing me with the FrameNet datasets. Thank you to Collin Baker for his help when I was new to FrameNet in my first year of candidature. Acknowledgement also goes to the NIST TREC community for their effort in providing researchers in the domain of information retrieval with a standardized benchmark and corresponding datasets without which it would have been very hard to train and evaluate

experimental QA systems and compare my work with the state-of-the-art systems in the field.

Furthermore, thanks to Dr. Yuval Marom from GAPbuster Worldwide and Dr. Timothy Baldwin from University of Melbourne for their useful comments on my research study.

Finally, I have to express my gratitude to my wife - Armita - for her great support along the way of my study. I cannot thank her enough for her love, patience, kindness, and taking care of our daughter and me while she was a student herself at the same time. And special thanks to my parents who have shown me how to grow even being so far from them. They have always motivated me and facilitated my study at different levels.

Thank you all again and I hope I will have a chance to return some of the immense support that I have received from you.

Contents

Statement of Originality **iii**

Source Material **v**

Abstract **vii**

Acknowledgements **x**

1 Introduction **1**

1.1 History of QA 3

1.2 Classes of QA Systems 5

1.3 Factoid QA 6

1.3.1 Pipelined Architecture of Factoid QA Systems 7

1.3.2 Some Challenges in Factoid QA 7

1.3.3 Linguistic Solutions to Factoid QA Challenges 9

1.4 Frame Semantics in FrameNet 10

1.5 Enhancing Factoid QA Using FrameNet 10

1.6 Contributions 11

1.7 Overview of the Thesis 12

2 Linguistic Approaches to Question Answering **14**

2.1 Passage Retrieval 14

2.1.1 Levels of Linguistic Knowledge 16

2.1.2 Passage Indexing 21

2.1.3 Online Analysis 22

2.1.4 Discussion of Key Aspects in Linguistic Passage Retrieval 24

2.2 Factoid Answer Processing 25

2.2.1 WordNet-Based Processes towards Answer Identification 26

2.2.2 PropBank in Answer Processing 29

2.2.3 FrameNet-Based Techniques to Answer Detection 32

2.2.4 Other Linguistic Resources . . . 37
2.2.5 Other Linguistic Structures . . . 38
2.2.6 Discussion of Key Aspects in Linguistic Factoid Answer Processing . . . 41
2.3 Research Problems . . . 44
2.3.1 Enhancing Answer Passage Retrieval for QA Using Linguistic Information . . . 45
2.3.2 Frame Semantic-Based Factoid Answer Processing . . . 45
2.4 Summary . . . 46

3 Methodology **48**
3.1 Answer Passage Retrieval in QA . . . 48
3.1.1 Enhanced Passage Retrieval Methods . . . 49
3.1.2 Data . . . 50
3.1.3 Baseline Passage Retrieval Systems . . . 51
3.1.4 Evaluation Metrics . . . 53
3.2 FrameNet-Based Factoid Answer Processing . . . 56
3.2.1 Experimental Setup for Evaluating FrameNet-Based Answer Processing . . . 57
3.2.2 Data . . . 60
3.2.3 Experimental QA System . . . 61
3.2.4 Baseline Shallow Semantic Parser . . . 67
3.2.5 Manual Annotation Tool . . . 67
3.2.6 Baseline QA Systems . . . 68
3.2.7 Evaluation Metric . . . 68
3.3 Summary . . . 69

4 Enhancing Answer Passage Retrieval for Question Answering **70**
4.1 Modifying MultiText . . . 70
4.1.1 Approach . . . 70
4.1.2 Experimental Results . . . 73
4.1.3 Discussion . . . 74
4.2 Frame Semantic-Based Retrieval Boosting . . . 76
4.2.1 Approach . . . 77
4.2.2 Experimental Results . . . 81
4.2.3 Discussion . . . 83
4.3 Summary . . . 84

5 The Effect of Levels of Frame Semantic Parsing on Answer Processing **86**
5.1 Related Work . . . 86
5.2 Levels of Frame Semantic Parsing . . . 87

5.3 Two-Step Gold Standard Annotation . . . 89
5.3.1 Approach . . . 89
5.3.2 Annotated Corpus . . . 90
5.3.3 Statistics of Annotation . . . 91
5.3.4 Quality of Annotation . . . 94
5.4 Experiments with Different Parsing Levels . . . 98
5.4.1 Initial Runs . . . 98
5.4.2 System Error Analysis . . . 99
5.4.3 Final Results . . . 102
5.5 Discussion . . . 103
5.6 Summary . . . 106

6 FrameNet-Based Answer Processing Techniques 108
6.1 FrameNet-Based Alignment Methods . . . 109
6.1.1 Complete Frame and FE Alignment - No Frame Scoring . . . 111
6.1.2 Frame Alignment with Specific FE Matching - No Frame Scoring . . . 113
6.1.3 Frame Alignment with Specific FE Matching - Frames Scored . . . 114
6.1.4 FE Alignment - No FE Scoring . . . 115
6.1.5 FE Alignment - FEs Scored . . . 116
6.2 Conceptual Analysis of Alignment Methods . . . 117
6.3 Predicate Chains and Complete Frame Semantic Alignment . . . 120
6.3.1 Ontologically Extended FrameNet . . . 121
6.3.2 Predicate Chain Representation using Extended FrameNet . . . 125
6.3.3 Reasoning on Predicate Chains for Answer Processing . . . 127
6.4 Experimental Results . . . 130
6.5 Discussion . . . 132
6.6 Summary . . . 134

7 FrameNet Coverage and FrameNet-Based Answer Processing 135
7.1 Linguistic Coverage . . . 136
7.1.1 Predicate Coverage . . . 136
7.1.2 Sense Coverage . . . 136
7.2 Naive Inductive Analysis of FrameNet Coverage . . . 137
7.3 FrameNet Statistics . . . 140
7.4 Practical Analysis of FrameNet Coverage and Factoid Answer Processing . . . 142
7.4.1 Experimental Results . . . 143
7.4.2 Discussion . . . 144
7.5 Summary . . . 145

8 Fusion of FrameNet-Based Answer Processing and Non-Semantic Approaches 146
8.1 Motivation . . . 146
8.2 Answer List Fusion Methods . . . 147
8.2.1 Rank-Based Fusion . . . 148
8.2.2 Score-Based Fusion . . . 150
8.2.3 Experimental Results . . . 151
8.2.4 Discussion . . . 152
8.3 Further Analysis on Score-Based Fusion . . . 155
8.3.1 Tuning Fusion Parameter . . . 156
8.3.2 Correct Answer Coverage . . . 159
8.4 Summary . . . 162

9 Conclusion 163
9.1 Recapitulation . . . 163
9.2 Contributions . . . 164
9.3 Epilogue: Frame Semantics Helps QA . . . 166
9.4 Future Directions . . . 167

A Extra Tables and Figures 170

B Acronyms 173

Bibliography 175

About the Author 189

List of Tables

2.1 Different levels of linguistic knowledge 16
2.2 A summary of the studies on query expansion/rewriting using linguistic or lexical knowledge 23
2.3 An example FrameNet frame 34

3.1 The filtering scheme of the experimental question sets 60
3.2 The usage of question sets to study the research problems 61
3.3 The mappings from question categories to NE types 63
3.4 SHALMANESER settings at each processing step 67

4.1 Accuracy of modified MultiText compared with those of MultiText and the Lemur passage retrieval methods on 208 TREC 2004 and 386 TREC 2006 factoid questions 73
4.2 The *mrr* values of modified MultiText compared with those of MultiText and the Lemur passage retrieval methods on 208 TREC 2004 and 386 TREC 2006 factoid questions 74
4.3 Average precision of modified MultiText compared with those of MultiText and the Lemur passage retrieval methods on 208 TREC 2004 and 386 TREC 2006 factoid questions 75
4.4 Average recall of modified MultiText compared with those of MultiText and the Lemur passage retrieval methods on 208 TREC 2004 and 386 TREC 2006 factoid questions 76
4.5 Probabilities (p-values after paired t-tests@10) obtained in the significance test between the results of modified MultiText, MultiText, and the Lemur passage retrieval methods on 208 TREC 2004 and 386 TREC 2006 factoid questions - first and second rows correspond to strict and lenient evaluations respectively - values with † are statistically significant ($p < 0.05$) 77
4.6 The methods selected for semantic boosting 82
4.7 Accuracy analysis of semantic boosting on 208 TREC 2004 and 386 TREC 2006 factoid questions 82

4.8 *mrr* analysis of semantic boosting on 208 TREC 2004 and 386 TREC 2006 factoid questions . . . 83
4.9 Average precision analysis of semantic boosting on 208 TREC 2004 and 386 TREC 2006 factoid questions . . . 83
4.10 Average recall analysis of semantic boosting on 208 TREC 2004 and 386 TREC 2006 factoid questions . . . 84
4.11 Probabilities (p-values after paired t-tests@10) obtained in the significance test between the results of the non-boosted method and its semantically boosted version on 208 TREC 2004 and 386 TREC 2006 factoid questions - first and second rows correspond to strict and lenient evaluations respectively - values with † are statistically significant ($p < 0.05$) . . . 84

5.1 Statistical information of the annotated data . . . 91
5.2 Average number of frames and FEs added/changed in manual correction - not-normalized measures . . . 92
5.3 Average number of frames and FEs added/changed in manual correction - normalized by the number of sentences . . . 92
5.4 Average number of frames and FEs added/changed in manual correction - normalized by the number of terms . . . 92
5.5 FrameNet-oriented statistics of the annotated data . . . 93
5.6 Parsing evaluations - retrieved passages from the AQUAINT collection . . . 93
5.7 Parsing evaluations - factoid questions from the TREC 2004 track . . . 93
5.8 An example frame agreement table for 10 predicates with four possible labels assigned by two annotators . . . 96
5.9 Inter-annotator frame agreement rates $\kappa_{(S\&C)}$ . . . 97
5.10 Inter-annotator FE agreement rates . . . 97
5.11 First QA runs on 143 TREC 2004 factoid questions - with Merged (FSB-ENB-fused) 98
5.12 First QA runs on 143 TREC 2004 factoid questions - with FSB-only . . . 99
5.13 TREC participant runs and our best run on 143 TREC 2004 factoid questions . . . 99
5.14 Error analysis on the frame semantic-based model of the experimental QA system . 100
5.15 Filtering the experimental dataset for experiments . . . 102
5.16 QA runs after finalizing the input question set (75 TREC 2004 factoid questions) - with Merged FSB-ENB-fused . . . 102
5.17 QA runs after finalizing the input question set (75 TREC 2004 factoid questions) - with FSB-only . . . 103
5.18 TREC participant runs and our best run on the selected 75 TREC 2004 factoid questions 103

6.1 Ontological relation set used to extend FN1.2 . . . 124

6.2 Instances of ontological relations over the FEs in FN1.2 124
6.3 Statistical information of the relation instances extracted for FN1.2 125
6.4 Part of the ontological extension on the frame "Accoutrements" between the FEs "accoutrement" and "wearer" at the end of the scenario 125
6.5 Mapping of ontological relations to basic plausible reasoning inferences 128
6.6 QA runs on 75 TREC 2004 factoid questions using the baseline system (the ENB-only setting) and the CFFE method of answer processing in the FSB-only setting 131
6.7 QA runs on 75 TREC 2004 factoid questions using the baseline system (the ENB-only setting) and the FSFE-NFS method of answer processing in the FSB-only setting . . 131
6.8 QA runs on 75 TREC 2004 factoid questions using the baseline system (the ENB-only setting) and the FSFE-FS method of answer processing in the FSB-only setting . . . 131
6.9 QA runs on 75 TREC 2004 factoid questions using the baseline system (the ENB-only setting) and the FE-NFES method of answer processing in the FSB-only setting . . 132
6.10 QA runs on 75 TREC 2004 factoid questions using the baseline system (the ENB-only setting) and the FE-FES method of answer processing in the FSB-only setting . . . 132
6.11 QA runs on 75 TREC 2004 factoid questions - combined settings are constructed by manual judgments of the two answer processing models - p-values after paired t-tests are calculated with respect to the Best-TREC system - values with † are statistically significant ($p < 0.05$) 134

7.1 The statistical information of a subset of the AQUAINT text collection on which an analysis of FrameNet coverage is conducted 137
7.2 All part-of-speech predicates not-covered after manual annotation with FN1.3 138
7.3 Verb predicates not-covered after manual annotation with FN1.3 138
7.4 Noun predicates not-covered after manual annotation with FN1.3 138
7.5 Adverb predicates not-covered after manual annotation with FN1.3 139
7.6 Adjective predicates not-covered after manual annotation with FN1.3 139
7.7 Preposition predicates not-covered after manual annotation with FN1.3 140
7.8 Part-of-speech-distinguished analysis of predicate coverage in our analysis sub-collection using FN1.3 - *all* occurrences 140
7.9 Part-of-speech-distinguished analysis of predicate coverage in our analysis sub-collection using FN1.3 - *unique* occurrences 141
7.10 Statistical information of the FN1.2 and 1.3 datasets 141
7.11 Coverage and growth ratio of different part-of-speech predicates in FN1.3 142
7.12 QA runs with different frame sets in different FrameNet datasets on 75 TREC 2004 factoid questions - values with † are statistically significant ($p < 0.05$) 143
7.13 QA runs with different FrameNet datasets used for training SHALMANESER on 75 TREC 2004 factoid questions - values with † are statistically significant ($p < 0.05$) . 143

7.14 QA runs with different FrameNet datasets used for training SHALMANESER on 176 TREC 2006 factoid questions - values with † are statistically significant ($p < 0.05$) . 144

8.1 The answer lists of the two answer processing models for the question Q44.2 in the TREC 2004 QA track retrieved by our experimental QA system 149
8.2 The answer lists and answer scores obtained by the two answer processing models for the question Q44.2 in the TREC 2004 QA track retrieved by our experimental QA system 151
8.3 The single answer list and answer scores after score-based merging for the question Q44.2 in the TREC 2004 QA track retrieved by our experimental QA system 152
8.4 QA runs on 75 TREC 2004 factoid questions with the rank-based fusion method, bold numbers show maximum values in each column 153
8.5 QA runs on 176 TREC 2006 factoid questions with the rank-based fusion method . . 154
8.6 QA runs on 75 TREC 2004 factoid questions with the score-based fusion method, bold numbers show maximum values in each column 155
8.7 QA runs on 176 TREC 2006 factoid questions with the score-based fusion method . 156
8.8 QA runs with different α values on 75 TREC 2004 factoid questions - FSFE-NFS method in the frame semantic-based answer processing model, bold numbers show maximum values in each column 157
8.9 QA runs with different α values on 75 TREC 2004 factoid questions - FSFE-FS method in the frame semantic-based answer processing model, bold numbers show maximum values in each column 158
8.10 *mrr* values for the individual answer processing models and their combinations using score-based fusion with $\alpha = 0.50$ on 75 TREC 2004 factoid questions 160
8.11 *mrr* values on 75 TREC 2004 factoid questions at important answer coverage points 161

A.1 Parameter set for document indexing using Lemur for the MultiText passage retrieval algorithm 170
A.2 Parameter set for passage indexing using Lemur 171
A.3 Parameter set for passage retrieval using Lemur 171

List of Figures

1.1 Two approaches of satisfying users' textual requests 2
1.2 Evolution of QA systems over the past decades 3
1.3 Example TREC 2004 question groups on the two topics of "Jennifer Capriati" and "space shuttles" 6
1.4 Pipelined architecture of factoid QA systems 7
1.5 A simplified "Sending" frame with four slots "sender", "theme", "medium", and "receiver" 9

2.1 Different syntactic parsing outputs of an example sentence, a) input sentence, b) part-of-speech view, c) grammatical parse view, and d) dependency tree view 17
2.2 Logical transformation in AnswerFinder; a) input sentence, and b) logical graph . . 40

3.1 Schematic view of study for linguistic passage retrieval in QA 49
3.2 The number of questions with at least a single correct answer on each passage rank (1 to 20) retrieved by our modified MultiText for factoid questions in the TREC 2004 and 2006 datasets 56
3.3 The hierarchy of the general activities to study the research problems in linguistic answer processing 58
3.4 Different experiments to address the factoid answer processing research problems - Experiment 2 is the main experiment of the thesis 59
3.5 The pipelined architecture of our experimental QA system 62
3.6 The question processing module takes three major steps 64
3.7 The flexible setting of the answer processing module 65
3.8 General scheme of frame semantic-based answer identification 66

4.1 Semantic boosting cycle of passage retrieval effectiveness 79
4.2 Schematic view of Example 4.3 81
4.3 Schematic view of Example 4.4 82

5.1 The three different facets of FrameNet-based annotation 88

5.2 Shallow semantic analysis of an example sentence evoking the frame "Manufacturing" 89
5.3 Incomplete automated shallow frame semantic parsing of an example sentence by SHALMANESER *before* manual correction 90
5.4 Comprehensive frame semantic annotation of an example sentence *after* manual correction 91
5.5 The scenario of analysing inter-annotator agreement on the annotated data 94
5.6 The different contribution rates to the *mrr* values in the Merged (FSB-ENB-fused) setting, fr: frame, FE: frame element, v: verb, nv: non-verb 105
5.7 The different contribution rates to the *mrr* values in the FSB-only setting, fr: frame, FE: frame element, v: verb, nv: non-verb 106

6.1 Different levels of evidence for answer candidate identification based on FrameNet; a) annotated passage region, b) annotated question region, c) the FEs in the match passage frame, and d) the vacant FE in the question frame 110
6.2 FrameNet-based alignment: A technical view 111
6.3 Identification of the question main frame 111
6.4 Frame and FE alignment in the CFFE method for answer candidate identification . 112
6.5 FE matching between a passage frame and a question main frame 113
6.6 Frame and FE alignment in the FSFE-NFS method for answer candidate identification 114
6.7 Frame score changing according to the existence of an instance value in the FE which corresponds to the vacant question FE 115
6.8 Boosting the score of passage frames according to the raw frequencies of query terms 115
6.9 FE alignment in the FE-NFES method for answer candidate identification - no frame matching is performed before FE alignment 116
6.10 FE scoring procedure in the FE-FES method 117
6.11 The level of matching elements and semantic information in the different FrameNet-based answer processing methods 117
6.12 A sample predicate chain; a) original text passage, and b) extracted predicate chain between the main entities of the passage 120
6.13 New dashed relations are not inferable in predicate chains with predicative relations 121
6.14 Entities and their relations; a) original predicate chain, and b) FrameNet-based mapping of the entities in the original predicate chain 126
6.15 Entities and their relations in a predicate chain - Ontological view 127
6.16 New extractable relations on a predicate chain; a) ontological view, and b) predicative view 128

6.17 Inference procedure for resolving predicate chains; C_1 and C_2 are certainty values of the inferences; a) question, b) answer passage, c) backward chaining, d) forward chaining, e) plausible notation of argument-based backward chaining, and f) plausible notation of referent-based forward chaining 129

7.1 Raw figures of *all* different part-of-speech predicates (in our analysis sub-collection) not-covered in FN1.3 139

8.1 The trends of strict answer processing *mrr*s for the two answer processing models and fused answer processing performance - FSFE-NFS method in the frame semantic-based model 159

8.2 The trends of strict answer processing *mrr*s for the two answer processing models and fused answer processing performance - FSFE-FS method in the frame semantic-based model 159

8.3 Correct answer coverage schemes by two answer processing models 160

8.4 Distribution of correct answers in Case 1 of Figure 8.3 161

A.1 Minimum number of samples required for estimating population proportion at the confidence level 95% and precision ± 0.03 172

Chapter 1

Introduction

Question Answering (QA) systems are natural language-based systems which people would naturally like to have to ask questions of and have responses from. These systems can ideally play the role of an oracle capable of answering any question related to any domain of knowledge. Such ultimate QA systems, like an oracle, can understand questions and have the capability of inferring logical answers from their knowledge-base. This knowledge-base is, therefore, supposed to cover all existing knowledge domains. However, it is not yet possible to formally code all human knowledge into machine understandable structures. As a result, one approach to QA has been to transform the process to an information retrieval-based process.

As the size, number, and type of information resources have grown, especially in the recent decades, the need for automated systems to conduct search processes in large amounts of information has emerged. As a result, the broad domain of information retrieval, in which search systems are generally classified, has been studied by many scholars. This field of knowledge covers all types of search for texts, images, and any other types of information and their combinations. In textual information retrieval, the process of search may look for text documents in document collections, small pieces of information in the documents (such as passages), or records of information in databases which are most related to an information need.

However, the idea of a QA system is to retrieve units that are very succinct and specifically related texts which directly answer a given question. This is addressed by extracting short text snippets from retrieved document or passage collections. Users dealing with QA systems do not need to worry about the formulation of their queries as if they were using traditional information retrieval systems (such as Google.com, Yahoo.com, and Altavista.com). This is because QA systems deal with natural language input information needs correctly formulated in a grammatical fashion which alleviates the burden of constructing the most informative keyword-based requests. Figure 1.1 shows the traditional information retrieval-based and the modern QA-based approaches of satisfying user requests. While in the traditional method texts are processed to extract the most informative indexes that are used

for query-text matching purposes, in the modern QA method the texts are comprehensively processed so that their linguistic structures are formulated in a machine-understandable format. With this meaning formulation methodology, it is possible to implement meaning-aware QA systems that retrieve focused responses to information requests rather than a large list of related documents. The main advantage of such brief responses is the elimination of the necessity of information seeking in a list of related text documents. In other words, the information seeker will not have to scan the text of a list of related documents in order to find the exact piece of information for which s/he is searching. For instance, the question "Who was the first woman in space?" will be succinctly answered by "Valentina Vladimirovna Tereshkova", instead of a stack of related text documents.

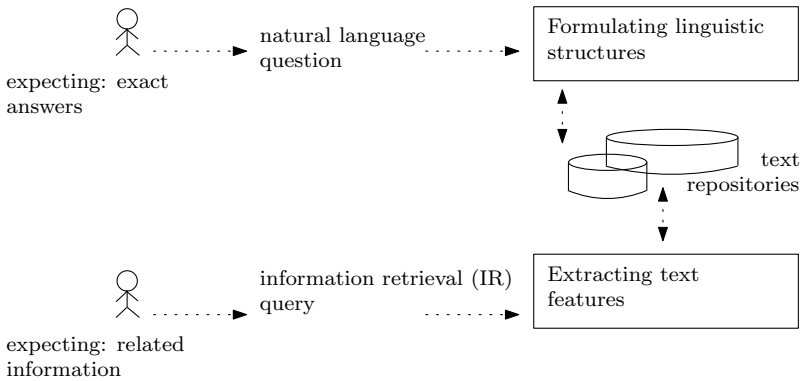

Figure 1.1: Two approaches of satisfying users' textual requests

In trying to capture important features that characterize the meaning of a text, a QA system may distinguish between different types of information requests. For example, the retrieved answer to a *when* question is a date or time which differs from the response to a *who* question that should be answered by the name of a person. This distinguishes QA systems from the traditional search engines which would remove question keywords such as *when, how, who,* and *why* as uninformative words. Traditional search systems generalize all such requests into a unique form which only contains information-bearing words and phrases. As a result, the list of retrieval units for both example questions "When was telephone invented?" and "Where was telephone invented?" would contain the same textual documents.

Although having sophisticated QA systems is an ideal, development of such systems is considered as an extreme problem in the domain of information retrieval. This is because of: i) the diversity of the text collections from which the answers are to be extracted, ii) the multiplicity of the types of the answer (knowledge) resources, iii) different text writing styles (syntax), and iv) different perspectives when composing texts (for example a journalist may write up an event in a different way compared to what may be written by a participant in that event). All these issues make the task of finding exact answers to the questions a very challenging problem. Another aspect of QA

systems that distinguishes them from other types of information retrieval applications is that a QA system should return exact and *specific* answers. This is the opposite to what is performed by the other information retrieval applications where the *relatedness* of retrieval units suffices.

1.1 History of QA

As shown in Figure 1.2, the first QA systems were developed as natural language interfaces for specific domains of knowledge in the early 1960s. One such system was the BASEBALL system that was capable of answering questions about the United States baseball league. As another instance, LUNAR (Woods, Kaplan, and Webber 1972) was also implemented to reply to natural language questions on the rocks returned from the moon by the Apollo moon missions. The knowledge resources for the preliminary QA systems over this period of time were mostly handwritten.

A few years later, by the end of the 1960s, there were other intelligent systems that included QA capabilities. SHRDLU (Winograd 1972) was one such system which could answer questions about different states in a Toy World. Basically, Toy World is a strategic planning method to move a few cubic blocks on a table and construct vertical stacks with different commands. Using each moving command at a time, only one block may be moved. This limits the movements so that the underneath blocks cannot be moved with one command.

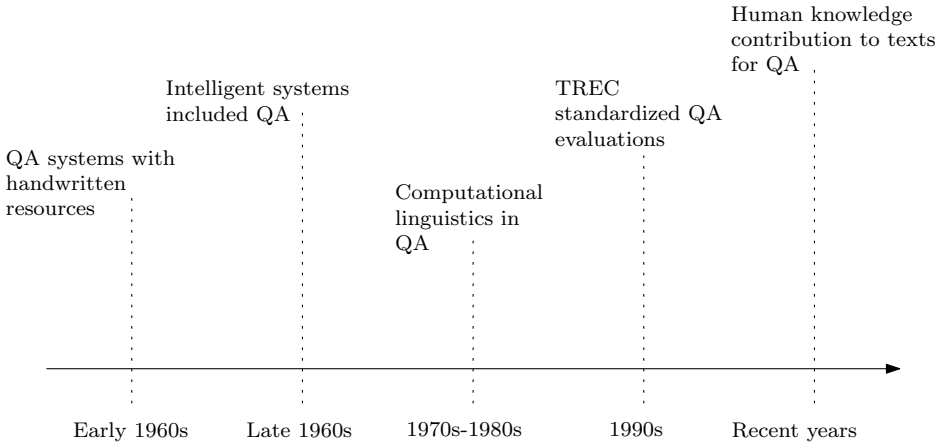

Figure 1.2: Evolution of QA systems over the past decades

Another famous system of this era was the well-known ELIZA system developed by Weizenbaum (1966). This psychological conversation provider enabled patients to converse with ELIZA as in an initial psychiatric interview. One of the intelligent aspects of ELIZA was to answer patients' statements with questions which were acceptable. For example, the question "What do you know about research in information technology?" would usually be replied to with "Does that question

interest you?". The system did not have any real world knowledge about any topic; however, it was designed to continue a conversation in a reasonable manner.

Later in the 1970s and 1980s, with the evolution of the domain of Computational Linguistics, QA systems were influenced drastically. The area of computational linguistics is a conjunction between two broad fields of knowledge, namely Computer Science and Linguistics. Computational linguistics covers different aspects of linguistics that can be handled by automated computer systems. The aim is to develop automated software systems capable of understanding the meaning of texts. This domain has made an invaluable contribution to the development of QA systems that require such understanding in the extraction of answers.

Unix Consultant (Wilensky 1982) and LILOG (Bosch and Geurts 1989) were two of the outstanding QA systems developed during the 1970s and 1980s. The former, also known as UC, has been designed to answer technical questions in the domain of the Unix operating system for computers. The LILOG system was capable of text understanding in the domain of tourism. It has been designed to reply to questions about tourism in a German city. Both UC and LILOG, unfortunately, were not maintained and released to the public; however, by being demonstrated they have nonetheless assisted the development of future similar efforts in the domain of QA.

In the 1990s, the Text REtrieval Conference (TREC) initiated the standard benchmarks for different tracks of information retrieval. The QA track of TREC has started to be one of the most standard evaluation and competition benchmarks in the QA domain. They provide a set of test questions and a text collection as the answer resource. Many QA systems from industrial and academic organizations compete with each other to answer questions that TREC provides every year. Best-performing systems are selected in each competition to present their QA approaches at the TREC conference.

In recent years, while TREC has still been active, there have been a variety of QA systems evolving on the basis of different knowledge-bases. They rely on human knowledge that can be added to texts. Such systems try to move towards a level of text understanding that can be achieved by humans using contextual information of texts and the real world knowledge achieved by experience. One best-known such system is the QA system developed at Language Computer Corporation[1] (LCC). LCC's PowerAnswer QA system (Moldovan et al. 2002) has been one of the most active participants in the TREC competitions. It was selected as the best-performing system in six consecutive competitions from 1999 to 2004. Recently, PowerAnswer has utilized the complementary tags of human knowledge on the texts of answer resources. These tags represent human knowledge underneath texts in a more explicit form that can be better interpreted by automated QA systems.

Another recently developed QA system is the START (SynTactic Analysis using Reversible Transformations) natural language QA system (Katz 1997) implemented at the artificial intelligence laboratory in Massachusetts Institute of Technology (MIT). START is a web-based QA system that can

[1] http://www.languagecomputer.com/

answer different types of questions on open-domain topics on the web. The key to the success of the START system is the offline process that it uses to annotate source texts of answers with advanced human knowledge. This ensures that texts are more machine-understandable. The knowledge representation process implemented in START differs from that articulated in the PowerAnswer system. Details of the two systems will be discussed in Chapter 2.

1.2 Classes of QA Systems

As QA systems have become continuously and increasingly complex, especially over recent years, there is now little in common across all such systems. Therefore, it is hardly possible to classify them into well-distinguished categories. Here, we focus on the *main* classes of QA systems that can be identified according to question types. This classification includes:

- **Factoid or fact-seeking questions:** Factoid QA systems respond to factual questions by returning a succinct piece of fact referring to the name of a person or place, the title of an organization, a date or time reference, a manner, or a reason. The question "In what year did France win their first soccer world cup?", for instance, is a factoid question the answer of which is the date or time reference "1998".
- **List questions:** The list QA systems respond to list questions with a list of facts. For example, the question "Which cities have Crip gangs?" seeks for a list of city names such as "New York, Chicago, and Boston". As there is no guarantee on the sufficiency of the answer list before exploring all of the related parts of an answer resource, the list QA systems require a thorough scan of all related sections to the question in the information resources.
- **Analytical questions:** The answer to such questions is not explicitly mentioned in knowledge resources and/or texts. Therefore, answering these questions entails comprehensive and deep inferential analysis on the knowledge elements of knowledge resources. For instance, the question "Which college is the oldest in North America?" necessitates retrieval of all related facts about colleges from the knowledge resource. In the absence of any explicit reference to the oldest college, the process continues with a comparative analysis between North American colleges according to their age and results in a final answer. In addition, the type of the answer to such questions is in most cases unanticipated. For example, for the question "What has been Russia's reaction to the U.S. bombing of Kosovo?" the answer might be diplomatic statements, behaviours, or decisions (Small et al. 2004).
- **Definition questions:** These questions are type-less questions the answers to which are sentences that define a certain concept. For example, the question "What is the Nobel Prize?" is a definition question that needs to be answered by a sentence like "The Nobel Prize is an annual award for outstanding contributions to chemistry...". Definition questions usually start with the question stem *What* and in order for them to be answered, a QA system needs to

carry out several types of text understanding, summarization, and reasoning processes.

Factoid and list QA systems have been studied in the TREC annual conferences, while analytical QA systems have not been adequately addressed so far. The TREC questions are grouped according to the target concepts represented by their identification number (Target ID) and an exact reference (Target string). As such, the following questions after the first question in each group may contain (anaphoric or non-anaphoric) references to the targets. Figure 1.3 shows two groups of questions in the TREC 2004 QA track containing three types of factoid, list, and *other* questions. The answer to other questions contains all related information to a specific topic which has not been covered by factoid or list questions in the related group of questions.

Target ID: 27
Target string: Jennifer Capriati

27.1: FACTOID: What sport does Jennifer Capriati play?
27.2: FACTOID: Who is het coach?
27.3: FACTOID: Where does she live?
27.4: FACTOID: When was she born?
27.5: OTHER: Other

Target ID: 65
Target string: space shuttles

65.1: LIST: What are the names of the space shuttles?
65.2: FACTOID: Which was the first flight?
65.3: FACTOID: When was the first flight?
65.4: FACTOID: When was the Challenger space shuttle disaster?
65.5: FACTOID: How many members were in the crew of the Challenger?
65.6: FACTOID: How long did the Challenger flight last before it exploded?
65.7: OTHER: Other

Figure 1.3: Example TREC 2004 question groups on the two topics of "Jennifer Capriati" and "space shuttles"

1.3 Factoid QA

Factoid QA systems extract succinct short focused answers to fact-seeking questions like "What is the name of the biggest moon of Saturn?". Factoid QA systems are known to be important to the extent that they extract factual knowledge from answer resources. Such answers can be beneficial in many applications, such as decision support systems, pedagogical and educational packages, business intelligence, and medical domain systems. The procedural methodology of factoid QA systems can be used in answering other types of questions (such as list questions). This emphasizes the significance of factoid QA. We, therefore, focus on this type of QA systems in this thesis.

1.3.1 Pipelined Architecture of Factoid QA Systems

A common pipelined architecture of factoid QA systems, as shown in Figure 1.4, consists of three main parts: i) question processing, ii) information retrieval, and iii) answer processing.

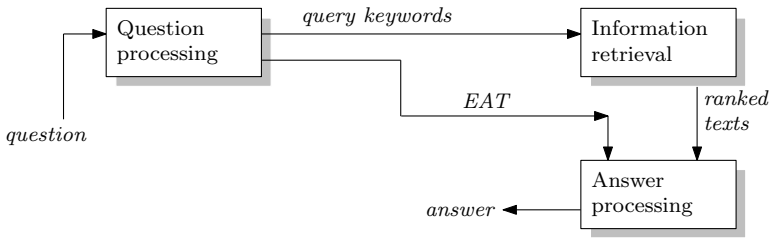

Figure 1.4: Pipelined architecture of factoid QA systems

The main tasks of the question processing module is to find the focus (required entity type) of a given question, known as an Expected Answer Type (EAT), and to construct an information retrieval query using the most informative keywords of the question. For the question "Why did Catherine commit suicide?" the EAT is a REASON and the information retrieval query is "Catherine commit suicide".

The information retrieval query is passed on to the information retrieval module where document-level, passage-level, or sentence-level information is retrieved. These textual units are those most related to the query which may contain potential and correct answers to the question.

The answer processing module[2] is designed to extract answer candidates from retrieved textual units, scoring answer candidates, and reporting top-ranked answers to end-users. This module receives the EAT from the question processing module which is used for filtering answer candidates according to question focuses[3]. The answer processing module deals with many text-related challenges in order to overcome surface (syntactical) mismatches between questions and textual units and to pinpoint potential answer-containing text spans.

1.3.2 Some Challenges in Factoid QA

In the different modules of the pipelined architecture of QA systems, there are many problems that affect QA performance. In the question processing part, identification of the features, which can be used to distinguish between different types of questions, has been one of the most difficult tasks. This is because there are different types of questions which share the same features. For example, the questions "What industry is Rohm and Haas in?" and "What kind of animal is an agouti?"

[2] Answer processing is a more general title for the task of answer extraction used in the literature and in this thesis. It includes answer extraction and scoring tasks.

[3] Some information retrieval modules also use the EATs for scoring or filtering documents or passages with respect to the main entities that questions ask for.

start with the question stem *What*; however, their EATs are *organization* and *animal* respectively. Therefore, a question processing module is required to consider more complex features than question keywords to differentiate between different question categories and to recognize corresponding EATs. Two main directions of question processing have evolved during recent years: i) rule-based analysis, and ii) learning-based classification. While the first direction focuses on the extraction of as many rules as possible to cover different types of questions, the second direction performs machine learning procedures to construct the best classifiers that can categorize questions into pre-defined classes.

In the information retrieval module, however, problems are more complicated as the retrieval engine must deal with extensive text-based challenges. One such challenge is the surface features of texts, which in many cases due to paraphrasing, do not match in different texts. For instance, consider the two text snippets:

> "In 1675, Cassini discovered that Saturn's rings are separated into two parts by a gap."

> "In 1675, it was found by Cassini that a gap divides Saturn's rings into two components."

The above sentences refer to the same event and similar concepts which are formulated in different syntactical structures. These syntactical structures can easily interfere with retrieval of many related and specific texts to an information retrieval query. In terms of QA, this problem is compounded because retrieved texts (documents or passages) are required to be exact in response to questions. Therefore, *relatedness* of the texts (to the questions) is not sufficient and they must be *specifically containing* answer candidates. For example, if the question asks "Who discovered the gap between Saturn's rings?" and the retrieved text is "In 1675, it was discovered that a gap divides Saturn's rings into two components", the retrieved text very much relates to the question; however, it does not contain any specific answer to the question.

In the answer processing module, many QA systems use different information extraction-based methods to discover exact answer spans in answer passages or documents. Most of information extraction-based methods use Named Entity (NE) extraction from passages to retrieve corresponding noun phrases to question categories as answer candidates. For instance, if the EAT of a given question is PERSON, then the NEs such as Michael, Kate, Floyd Patterson, and Huygens can be retrieved as answer candidates from specifically related passages.

In cases where the EAT is a REASON or MANNER, however, the information extraction-based methods can hardly extract answer candidates. This is sometimes because the answer candidates may contain none or more than one type of NEs.

It is also possible that there are a number of redundant NEs in answer passages. This makes the task of answer processing shallow and semantically unaware which cannot be performed at a high level of confidence. For example, the answer passage "In 1958, Jack started his journey from Chicago to Paris." for the question "Where did Jack travel to in 1958?" may easily confuse the answer processing procedure with the two existing LOCATION references "Chicago" and "Paris".

1.3.3 Linguistic Solutions to Factoid QA Challenges

Natural language-based approaches in QA can be exploited to linguistically *resolve* or *boost* factoid QA modules. Different linguistic approaches have been used to overcome information retrieval problems and answer processing challenges by adding linguistic information to the text of questions and/or documents and passages. Linguistic information can be in the form of syntactical attributes, morphological constructions, and semantic features. While the usage of these types of information in QA has been studied much during the past years, the contribution of scenario-based or scenario-based association information has not been carefully researched yet.

Scenario-based information can be encapsulated in *semantic frames* with slots representing participant roles and frames containing the whole scenario of an event or state. The information which fills the slots, the *filler*, is not a frame in this definition since the participant roles in an event or state are not events or states themselves. This is a major difference between these semantic frames and those introduced by Minsky (1974). Figure 1.5 shows an example frame that symbolically formulates a simplified version of the event "Sending" with four participant roles "sender", "theme", "medium", and "receiver".

Each semantic frame can cover a list of target words or predicates that share the same semantic features (event or state definition and participant roles). The predicates "send", "ship", and "export" are some instances that are inherited from the semantic frame of "Sending".

The structure of semantic frames allows retrieval of a greater number of specifically related passages to natural language questions. Such related passages cannot be reached by using other types of semantic information like those formulated in different words with the same meaning (synonyms) or conceptually more general words (hypernyms) or more specific words (hyponyms). For instance, retrieval of the sentence "X, son of Y, was the first person on the moon" for the question "Who was X's mother?" can only be achieved when considering the scenario of "Kinship" in a semantic frame from which the noun predicates "son" and "mother" are inherited.

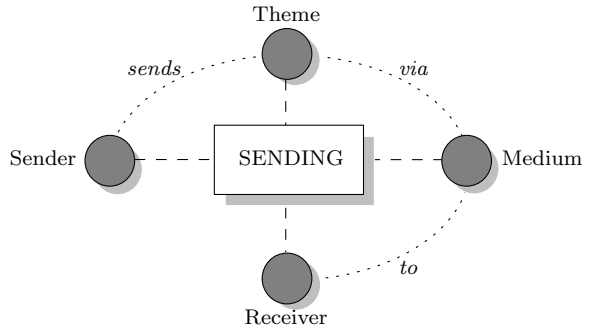

Figure 1.5: A simplified "Sending" frame with four slots "sender", "theme", "medium", and "receiver"

To resolve or boost answer processing in cases where information extraction-based approaches fail

or are not very confident, frame semantic information can assist to identify previously unaccessible answer spans or select answers that match with the certain semantics of a given question. This is addressed by semantic alignment of semantic frames and their slots in questions and answer-containing passages. Semantic alignment of the question "What did he die of?", for example, with the answer passage "He died of kidney failure while filming in San Francisco." can be performed between the same semantic frames "Death" and the slots "protagonist" and "cause" in both texts. As a result, the answer span "kidney failure", as the filler of "cause", can be extracted. This is not viable using many information extraction-based techniques and other procedures which use other types of linguistic information. The question "Where did Jack travel to in 1958?", mentioned in section 1.3.2, with the answer passage "In 1958, Jack started his journey from Chicago to Paris." can also be handled by aligning the semantic frame "Travel" and the slots "traveler" and "goal". Since "Paris" is the filler for the slot "goal" in the passage semantic frame, the answer processing module can confidently discard the other answer candidate "Chicago".

1.4 Frame Semantics in FrameNet

A type of frame semantic information, referred to as Frame Semantics (Fillmore 1976; Lowe, Baker, and Fillmore 1997; Petruck 1996), has been developed in recent decades which emphasizes the continuities between language and human experiences. Frame semantics has been encapsulated in FrameNet (Baker, Fillmore, and Lowe 1998), which is a network of inter-related semantic frames. The main advantage of using FrameNet frames compared to other types of semantic frames such as those in PropBank (Kingsbury and Palmer 2003; Palmer, Gildea, and Kingsbury 2005) and VerbNet (Schuler 2005) is that different part-of-speech predicates (nouns, verbs, adjectives, adverbs, and prepositions) can be covered in a single FrameNet frame as opposed to verb-based semantic frames in PropBank and VerbNet. The details of the structure of FrameNet will be discussed in Chapter 2.

In this research, we focus on the semantic frames of FrameNet, thus, by frame semantic-based procedures in QA, hereafter, we refer to the processes which use the specific type of frame semantics encapsulated in FrameNet.

1.5 Enhancing Factoid QA Using FrameNet

One of the major challenges of modern QA research is that the end-to-end performance of QA systems is evaluated instead of any component-level analysis. This leads to no significant understanding of the underlying techniques used in each component. The central topic of this thesis is to investigate and contribute to two parts of factoid QA systems - the information retrieval and answer processing modules. We use linguistic information especially the frame semantics encapsulated in FrameNet.

In the case of the information retrieval part, we will study ways of improving answer passage

retrieval performance. We exploit different types of linguistic and non-linguistic information underlying the surface structure of the texts of questions and passages. These information types include the traditional density-based[4] and syntactical information of query terms, topical information of queries, the length of passages, the rate of covering query terms by passages, and more importantly the scenario-based relations between query and passage terms. By using these types of information, we will investigate ways of retrieving a greater number of answer passages in a short list of retrieved passages. This will also include improving the rank of answer passage in short sorted lists of retrieved passages.

We will also study different aspects of articulating the semantic information which FrameNet provides in the task of factoid answer extraction and scoring. Our work suggests solutions to exploit the FrameNet linguistic resource through achieving an increased factoid answer processing performance. The underlying difficulty of semantic parsing (to add scenario-based information to texts) is considered by investigating the effects of different levels of shallow semantic parsing (that can be achieved based on FrameNet frames and frame elements) on the outcomes of this work. We will also study the possible benefits of fusing FrameNet-based answer processing techniques with other non-semantic models of answer extraction and scoring.

To follow a standard benchmark on factoid QA, the TREC QA track is used in this thesis in conducting experiments and analysing research questions in both information retrieval and answer processing parts. The detailed explanation of the research problems studied in this thesis will be given in section 2.3.

1.6 Contributions

Our work contributes useful and new knowledge to the domain of FrameNet-based QA, which is a novel approach in tackling challenges in factoid QA already discussed in section 1.3.2. We will show how event-based (or state-based) associations between terms in the text of questions and answer passages can assist in enhancing factoid QA performances.

- In the passage retrieval phase, we will show that:
  - The usage of linguistic and non-linguistic (topic-based and keyword-based) information in scoring and raking passages retrieved for natural language factoid questions results in an improved answer passage retrieval performance compared to a number of existing passage retrieval methods.
  - The retrieval of a greater number of answer passages with high ranks is possible by using scenario-based relations between question and passage terms. These relations can be extracted from appropriate FrameNet frames that cover question predicates.

[4] Term density-based information may include term frequency, term proximity, and term coverage measures.

- In the answer processing phase, we will conclude that:
  - The answer processing performance of factoid QA systems which use FrameNet frames is proportional to the level of accuracy in shallow semantic parsing. More interestingly, different part-of-speech predicates play different roles in enhancing this performance. We will show that non-verb frames are almost as important as verb frames in this regard.
  - There are a number of FrameNet-based techniques of answer processing implemented in our work which suggest that strictly tying question and passage frames (considering all participant roles) does not offer high answer processing performance. Instead, our new relaxed approach which considers query context and certain participant roles under question performs best.
  - Linguistic coverage of different part-of-speech predicates (in FrameNet) affects factoid QA performances. We will show that the coverage of noun predicates is in a crucial situation at this stage and covering a greater number of nouns in FrameNet is more important in enhancing answer processing performance compared to other part-of-speech predicates.
  - The hybridization of FrameNet-based answer processing models with non-semantic models can be more effectively done by using linear functions of merging answer lists (of the two types of models) compared to the approaches which do not consider sophisticated answer list merging strategies and treat all answers with equal weights.

1.7 Overview of the Thesis

Chapter 2 will disclose the literature of linguistic approaches to factoid QA regarding the two parts of factoid QA systems namely passage retrieval and answer processing. In Chapter 3, the methodological aspects of the thesis will be explained. This includes the explanation of the required settings, baseline systems used in our study, and the evaluation metrics on which the results of our experiments are evaluated and compared with those of other studies. Chapter 4 investigates the specific research questions on the information retrieval part of factoid QA. This includes our contributions on using semantic and non-semantic information for retrieving a greater number of answer-containing passages. In Chapter 5, we study the effect of different levels of assigning semantic scenario-based information to texts on the answer processing performance using FrameNet. For doing this, we consider a number of semantic parsing levels on a manually corrected annotated corpus. Chapter 6 studies a range of different FrameNet-based answer extraction and scoring techniques. In this chapter, we introduce our new frame semantic-based answer processing techniques one of which outperforms other new and existing methods. The effect of the lexical coverage of FrameNet on the performance of answer processing task is studied in Chapter 7. To observe the effectiveness of FrameNet-based answer processing methods in conjunction with non-semantic approaches, in

Chapter 8, we analyse the overall and individual performance of a FrameNet-based and an entity-based answer processing models when their results are fused with each other using two answer list merging methods. Finally, Chapter 9 concludes this thesis with the main results obtained in this research followed by some directions for future work.

Chapter 2

Linguistic Approaches to Question Answering

The usage of linguistic knowledge in QA systems has been considered by many researchers in recent years. There are studies which conclude that the wise exploitation of such knowledge can improve the effectiveness of the QA task (Bernardi et al. 2003; Cardie et al. 2000; Harabagiu, Paşca, and Maiorano 2000). In a typical pipelined QA architecture, such improvement can be obtained within the different sub-tasks of question processing, information retrieval, and answer processing each of which needs careful consideration in a linguistically aware QA system. While the main outcomes of question processing is identification of the focus of the questions or EATs, and information retrieval queries, the output of the information retrieval part is a list of related documents or passages to the questions that may contain the actual answers. The main task in the answer processing phase is to identify answer candidates from the related documents or passages with respect to the EATs and score and rank them.

In this chapter, we focus the review on the existing approaches in the two main phases of information retrieval - more specifically passage retrieval - and answer processing (answer extraction and scoring). The emphasis of the chapter is on the linguistic approaches in the two phases. The analytical discussions conducted with respect to each phase justify the two main concerns of the research problems in this thesis.

2.1 Passage Retrieval

One possibility for the successful extraction of candidate and actual answers is to use parts of the texts which are most similar to the concept(s) of the information sought. There are empirical studies in the domain of QA which show that the answer processing task can be handled more effectively on the passage-level information in documents rather than document-level texts (Clarke et al. 2001; Clarke and Terra 2003; Harabagiu and Maiorano 1999; Lee, Hwang, and Rim 2002; Moldovan et al. 2003*b*; Oh, Myaeng, and Jang 2007). It has become apparent to these researchers that handling the

task of answer processing in the more specifically related short passages, where the answers may be found, is more effective than searching through the whole text of related large documents.

In the context of QA, there are challenges which arise in trying to identify succinct passages that potentially bear the answer candidates to a given question. That is, *relatedness* of the passages alone may not be the best criterion based on which to train and evaluate the retrieval systems. Therefore, *specificity* is more desirable when trying to ensure that passages contain answer candidates. This creates new problems and requires more precise text understanding processes in order to attain more effective QA systems.

With this in mind, the work in (Roberts and Gaizauskas 2004) studies the shortcomings of the traditional information retrieval-based precision and recall measures to evaluate passage retrieval systems. The two new measures that they introduced were *coverage* and *redundancy*. The former formulates the proportion of a question set for which at least one correct answer can be found in the top n passages that are retrieved per question. The latter shows the average number of passages within the top n passages which contain a correct answer for each question. The redundancy measure reflects the chance for an answer processing module to successfully identify a correct answer and introduces an important evaluation metric in this respect.

There has been a lot of work in the area of passage retrieval using different techniques and knowledge resources since its emergence. From a big picture point of view, there are two broad directions in which the passage retrieval task (for QA) can be conducted: i) the *general non-linguistic* approaches, and ii) the *linguistic* approaches. The main difference of the two directions is in the type of external and/or intrinsic knowledge resources that they use to formulate input queries, identify the most related text snippets, and score and rank the passages according to a similarity function between the queries and passages.

In the case of the general non-linguistic approaches to passage retrieval, most of the methods only rely on the statistical density-based information about term occurrences in passages as in (Clarke, Cormack, and Burkowski 1995; Clarke et al. 2000; Clarke, Cormack, and Tudhope 2000; Cormack et al. 1998; Hovy, Hermjakob, and Lin 2001; Llopis and Vicedo 2001; Mittendorf and Schauble 1994; Robertson, Walker, and Beaulieu 1998; Vicedo and Ferrandez 2001), the Pauchok implementation of the SiteQ's passage retrieval algorithm (Lee et al. 2001), and others. Although these methods perform well and represent the state-of-the-art in retrieving *related* passages to generic information retrieval queries, in the context of QA, they cannot overcome many text-based challenges such as the surface mismatches between linguistically related concepts and terms to retrieve *specific* passages containing actual answers to a given natural language question. This problem is now recognized as one of the most noticeable difficulties preventing QA systems from improving their effectiveness (Vicedo 2001).

In this thesis, therefore, we focus on linguistic passage retrieval where the techniques mainly rely on the syntactic, morphological, and semantic analysis of the texts of the document collections and

information requests. We conduct a review of some of the most-cited existing studies related to the problems considered in this thesis.

2.1.1 Levels of Linguistic Knowledge

The first aspect that must be recognized in a linguistic passage retrieval process is the level of the linguistic knowledge and the resources that are referred to by the retrieval system in order to perform any level of surface or meaning-oriented analysis of texts. Table 2.1 shows the taxonomy of the levels of linguistic analysis that are generally taken into consideration in a combined manner in many natural language applications. Detailed explanation on these levels can be found in the literature of theoretical and computational linguistics.

Table 2.1: Different levels of linguistic knowledge

Linguistic level	Focus
Phonetics	Production and perception of speech sounds
Phonology	Organization of linguistic sound patterns
Morphology	Context-based word shapes and behaviours
Syntax	Structural relations
Semantics	Meaning and lexical relations
Pragmatics	Manner of exploitation of language to achieve desired goals and effect of context on meaning
Discourse	Consideration of coherent sequences of texts

In the following sections, we review the works that have utilized syntax, morphology, and semantics to overcome different intrinsic challenges in answer passage retrieval.

2.1.1.1 Syntax

The syntactic analysis of the texts, to pinpoint the structural features underlying the texts, is widely dependent on the automated syntactic parsers which can generate syntactic tags at different levels. Part-of-speech tagging, grammar representation, syntactic function analysis, dependency relations extraction, and recognizing syntactic patterns are the most common syntactic analyses that are formulated in different methods. Figure 2.1 shows an example sentence syntactically parsed with some different views, with the details skipped as being out of the scope of this chapter.

Syntactical features have been used by many researchers as the basic starting point for more sophisticated morphological or semantic analyses in the domain of information retrieval. The exploitation of part-of-speech tags can be found in SiteQ's passage retrieval algorithm (Lee et al. 2001) for weighting query terms and sentences, and in (Clarke et al. 2000) in conjunction with other grammatical and question keyword information for adding extra words to the representative query for passage retrieval.

Dependency trees are used in (Sun, Ong, and Chua 2006) to overcome the statistic-based term

a)
She is a new PhD student at the University of Ballarat.

b)
She/PRP is/VBZ a/DT new/JJ PhD/NN student/NN at/IN the/DT University/NNP of/IN Ballarat/NNP ./.

c)
```
(ROOT
  (S
    (NP (PRP She))
    (VP (VBZ is)
      (NP
        (NP (DT a) (JJ new) (NN PhD) (NN student))
        (PP (IN at)
          (NP
            (NP (DT the) (NNP University))
            (PP (IN of)
              (NP (NNp Ballarat)))))))
  (. .)))
```

d)

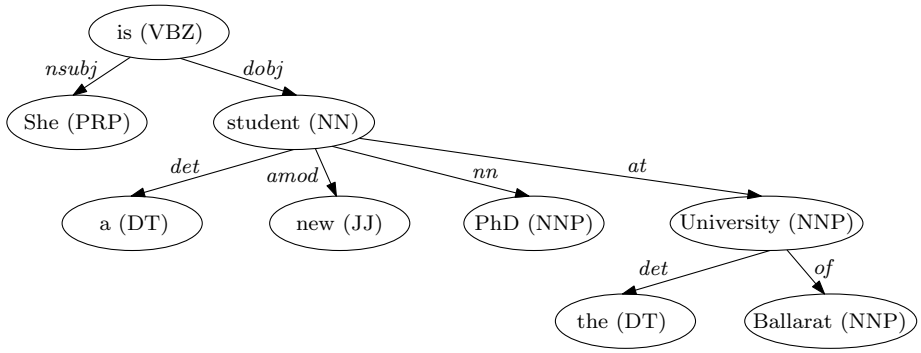

Figure 2.1: Different syntactic parsing outputs of an example sentence, a) input sentence, b) part-of-speech view, c) grammatical parse view, and d) dependency tree view

co-occurrence analysis in the Local Context Analysis (LCA) approach of query expansion (Xu and Croft 1996), in (Tiedemann 2005) using a deep syntactic dependency parser for Dutch called Alpino, in (Harabagiu et al. 2001) for identifying the semantic forms of questions and answer paragraphs, in (Cui et al. 2005) in a fuzzy relation matching procedure for matching question and passage structures, and in (Kaisser 2005; Kaisser and Becker 2004; Kaisser, Scheible, and Webber 2006; Kaisser and Webber 2007) for identifying the shared paths to head verbs and answer roles and finding answer sentences. The last works use other linguistic structures of sentences like sequences and targets explained in (Kaisser and Becker 2004) in conjunction with dependency trees.

There are other works that use syntactic information for more effective answer passage retrieval. These include the AnswerFinder QA system (Molla 2003; Molla and Gardiner 2004; Molla and Gardiner 2005; Molla, Zaanen, and Pizzato 2006) that uses grammatical relation overlaps between questions and sentences as one of the criteria for scoring single-sentenced passages, the work in (Choquette 1996) for more comprehensively analysing compound terms, the work by Woods et al.

(2000*b*) that uses syntax information in a lexicon for the analysis of phrases along with the semantic and morphological analyses, and the semantic analysis of questions in (Vicedo 2001) that starts with the representation of the semantic content of the syntactic structures in the questions.

All of the above works suggest improved answer passage retrieval effectiveness for QA systems when using syntactic information.

2.1.1.2 Morphology

At the level of morphological analysis, different shapes and formations of a single term are analysed. Understanding of the similar inheritance between the two words, for instance "hand" and "handcraft", is realized at this level of linguistic analysis which studies the internal structure of words.

In different efforts to enhance the effectiveness of the task of passage retrieval, linguistic knowledge at the level of morphology has been used especially with the aim of developing shape relaxations over surface mismatches of the texts. In (Neumann and Sacaleanu 2004), natural language generation by morphological analysis of the terms is used to expand passage retrieval queries in the domain of QA. Morphological variations of the terms are also considered for calculating the term similarities between questions and document sentences in the QALC QA system (Ferret et al. 2000) to retrieve and rank the most similar sentences from the document corpus. Offline conceptual indexing in (Woods et al. 2000*b*), as indicated above, uses the morphological rules to extend a core lexicon with more entries and to perform different types of relaxations on the terms in the queries and passages and overcome the surface mismatches between them. It also performs some morphological analysis of unseen compound words. Another method of overcoming paraphrasing problems is used in retrieving related passages to the TREC relationship questions (Katz et al. 2005; Marton and Katz 2006). It takes into account some word and phrase-level variations in the texts of the questions and answer-containing passages using features such as morphological ones. The morphological alternatives of question keywords are also used as one of the feedback loops to boost the accuracy of the paragraph selection process in (Harabagiu et al. 2001).

2.1.1.3 Semantics

The main goal of the semantic analysis of texts is to suggest beneficial semantic normalizations[1] over the meaning of differently expressed similar concepts in the texts. While syntactic analysis reveals the structural features and morphological analysis pinpoints the different possible formations of terms, semantic analysis moves towards meanings hidden behind the surface properties of the texts; therefore, it can develop more human-like text analysis and understanding.

The morphological analysis in (Ferret et al. 2000) is accompanied by a semantic analysis of the

[1] Semantic normalization refers to a process in which different lexical units - predicates - are grouped with respect to pre-defined significant semantic features.

terms to find the similarity measures between the questions and document sentences. The variants of the terms are extracted by FASTR (Jacquemin 1999) which is a two-tier model describing morphological, syntactic, and semantic variations of terms exploiting five different sources including WordNet (Miller et al. 1990). The semantic relation taken into consideration in this study is synonymy. The WordNet synonymy relations are also exploited in (Yang and Chua 2002) in conjunction with the WordNet glosses and the Web local context to expand queries for more focused sentence retrieval and ranking. This work tries to combine lexical knowledge and external resources in order to overcome the gaps between the query space and document space.

In (Woods et al. 2000*b*), the authors use the semantic relations such as "kind-of", and "instance-of" to relate more specific concepts to more general concepts in a conceptual taxonomy. Then, such information is used for retrieving more related passages in a penalty-based scoring methodology to take the effect of the relaxations into consideration. Their retrieval system, called Nova, is then exploited in their QA system which participated in the TREC 2000 QA competition (Woods et al. 2000*a*). With the retrieval of 5 answers per question in the TREC 2000 QA track, their system was able to answer almost half of the questions in the test set.

Shallow information extraction-based knowledge (NE labels) used in (Tiedemann 2005), along with other types of linguistic information, elevates the performance of the passage retrieval task compared with a baseline system (which uses plain text keywords only) by 15% in mean total reciprocal rank. This approach has the potential for the exploitation of more sophisticated semantic features.

The study conducted in (Vicedo 2001) represents the semantic content of the question terms, referred to as syntactic structures, by exploring the synonyms and one-level-search hyponyms and hypernyms extracted from the WordNet lexicon. By constructing a semantic content vector per question concept, their approach develops a semantic normalization that can cover different ways of semantic representation of the different concepts appearing in the text of a given question. Their approach results in both new answer finding and ranking improvement over a baseline system that does not consider the semantic content.

WordNet-based synonyms and hypernyms are also used in (Harabagiu et al. 2001) to reformulate paragraph retrieval queries when there are no answers found in the passages retrieved by the original query. A similar approach is used in (Hovy et al. 2000) to expand queries for retrieving documents which are consequently segmented into topical sub-parts (passages). In this indirect passage retrieval task, their expansion process benefits from WordNet to find more retrieval keywords. The main variations taken into consideration to retrieve answers to relationship questions in TREC (Katz et al. 2005; Marton and Katz 2006), in addition to the morphological features, are based on the NOMLEX structures (Macleod et al. 1998), the Wikipedia synonyms, and the variants from a small manually compiled thesaurus. With these features, their retrieval technique identifies the semantic overlap of the questions and passages and scores the passages accordingly.

IBM's passage retrieval algorithm (Ittycheriah, Franz, and Roukos 2001; Ittycheriah et al. 2000) also uses WordNet synonyms of the query terms which appear in the passages as one of the measures to score and rank the passages. The shallow semantic-based approach in (Moreda, Navarro, and Palomar 2005) tries to improve the effectiveness of a passage retrieval system, IR-n (Llopis and Vicedo 2001), by exploiting semantic roles in the semantic frames of PropBank (Kingsbury and Palmer 2003; Palmer, Gildea, and Kingsbury 2005). They try to initiate work to make use of the predicate-argument structures of the passages as a metric for measuring the semantic similarity of a given information request and the passages. Their experimental results are yet to be published.

The syntactic analysis of questions and answer sentences in AnswerFinder, as mentioned before, is accompanied by semantic analysis to extract the Flat Logical Forms (Molla 2001) and Logical Graphs (Molla and Gardiner 2005) in order to find the semantic similarity between a given question and the answer sentences. The semantic similarity between the question and sentences using the flat logical forms is calculated using the number of the logical terms that occur in both sides. This requires the analysis of the overlaps between the logical graphs, which will be discussed in section 2.2.4.

The shallow semantic parsing-based approach in (Hickl et al. 2006) considers the distribution of the question frame semantics structure (Fillmore 1976; Lowe, Baker, and Fillmore 1997; Petruck 1996) in retrieved passages for further scoring and ranking of the passages. This develops a scenario-based mechanism for estimating the semantic similarity of the questions and passages. There are no passage retrieval-based results available from their complex QA system as their focus is on the overall performance of answering questions. In (Schlaefer et al. 2007), hypernymy, hyponymy, meronymy, holonymy, and synonymy relations of WordNet are used to expand passage retrieval queries. They also use PropBank-based analysis of questions to generate predicate queries for passage retrieval in the context of factoid QA. Again, they do not report on the passage retrieval performance of their QA system.

One of the recent works that more strongly demonstrates semantic analysis in the context of passage retrieval can be found in (Oh, Myaeng, and Jang 2007). It studies an open domain passage retrieval method which is exploited in the context of QA. The method considers variable-length passages that are constructed dynamically on the basis of semantic information in the sentences which formulate their topics. To construct the semantic passages, the first step is to classify the sentences of the documents into predefined taxonomical classes inter-related via *is-a* relations semi-automatically devised by the authors. The learning-based classifier takes sentence patterns of shallow semantic information such as the entity type of the nearest neighbour noun to the verbs. It also considers lexical extensions of the verbs in the forms of synonyms, hypernyms, and hyponyms. Once the sentences have been classified into the topics, they are grouped according to their topic labels. In the online retrieval scenario, different features of the constructed and indexed passages and questions are taken into account such as the question title, question keywords, passage topic, and answer type.

The first important aspect of this study is the capability of the system to develop semantic passages using linguistic features from any position in the documents containing topically related sentences. The second noticeable direction in this work is the semantic normalization that is conducted in the retrieval process by exploiting different semantic features of the passages and questions to improve the retrieval task through a more meaning-aware process. Their experiments show that their topic-based semantic passage retrieval is more useful than the fixed-length passage retrieval in the context of QA.

The semantic approaches to improve passage retrieval effectiveness have been applied in restricted knowledge domains as well. The concept-based approach demonstrated in (Zhou et al. 2006) on biomedical HTML full-text documents can be referenced as an instance. In order to identify the most related passages to a given biomedical query, this study identifies the similarity between the paragraphs and passages to the query using both the concept and word similarity features. The emphasis, however, is on the concept similarity feature which is calculated by taking into consideration the semantic variations of the biomedical terms. In general, their study with semantic information features shows an improvement over the baseline retrieval system where no such features are contributed to the retrieval process. Another attempt in restricted domains can be found in (Lin and Demner-Fushman 2006) which benefits from the semantic knowledge to leverage the retrieval performance in the domain of clinical medicine. However, their study is more concerned with information retrieval at the level of textual documents. In (Stokes et al. 2007), the synonyms, hypernyms, and hyponyms (of diseases and biological process mentions) are used for expanding answer passage retrieval queries. This work reports improvements in genomic information retrieval performance.

2.1.2 Passage Indexing

Passage indexing, similar to the task of document indexing, is one of the approaches that most existing general and linguistic passage retrieval systems exploit in order to directly access the passage-level information in the documents eliminating the process of online passage boundary detection. Different methods implemented in the Lemur retrieval package[2] all work on the basis of such indexes.

To use linguistic knowledge in enhancing the effectiveness of the passage retrieval task, offline passage indexing has been selected as a solution. The linguistic knowledge can contribute to the linguistic coverage of the passages and such information can be stored in an offline index to be exploited in the online retrieval task. The main advantage of this approach is to reduce the latency of the extensive linguistic-oriented query and passage analysis and evaluation and to enhance the efficiency of the systems in line with improving the effectiveness that is sought. The method demonstrated in (Woods et al. 2000*b*; Woods et al. 2000*a*) is based on the conceptual indexing approach that benefits from syntactic, morphological, and semantic information. Indexed material is connected to a conceptual taxonomy which plays the role of the linguistic knowledge resource.

[2] http://www.lemurproject.org/

The multi-layer index, constructed in (Tiedemann 2005) for Dutch texts, contains three main layers: i) token layers, ii) type layers, and iii) annotation layers. The token layers include features such as plain text tokens and root forms. The type layers include specific types of tokens like named entities and compounds. The annotation layers can only contain the labels of token types such as LOCATION, PERSON, and ORGANIZATION. The multi-layer index can be accessed at each layer with corresponding appropriate restrictions that may be imposed according to the limitations caused by a given question. For instance, it is possible to query the index only by formulating root forms to be matched with the root layer of the token layers. As there are many possible combinations of restrictions over the features, the author tries to best formulate a query after applying a genetic algorithm-based optimization technique to train their system on finding the best restriction and weighting schema for the query linguistic features.

The semantic segmentation of texts in (Oh, Myaeng, and Jang 2007) results in constructing and indexing semantic passages according to sentence topics. Their study also considers some online analysis of the questions in order to retrieve the most related passages to the topic of a given question. The semantic indexing introduced in Sapere (Katz and Lin 2003) using the ternary expressions (Katz 1990) is another demonstration of offline indices exploited in retrieving the most specifically related text snippets to a given information request. Sapere stores all such triplet relations of the resource text in the form of SUBJECT-verb-OBJECT including passive constructions, adjective-noun modification, noun-noun modification, possessive relations, predicate nominatives, predicate adjectives, appositives, and prepositional phrases. Although the approach taken by Sapere converges to the direct answer selection for a QA system easily, it can still be categorized as a linguistic index-based passage selection method that identifies specific text snippets for answer processing purposes. They have compared their Sapere system with a Boolean baseline passage retrieval system with sentence-level indexing. The Sapere methodology drastically outperforms the Boolean system as reported in (Katz and Lin 2003).

2.1.3 Online Analysis

In contrast with the offline passage indexing techniques, there are techniques that approach more effective retrieval of passages by performing online analyses on the queries or questions and the texts of the passages. Such techniques, in the presence of the complexity of the linguistic processes, suffer from deficiencies such as being time-consuming. However, their effectiveness may reach higher levels by adopting more sophisticated question or query concept-oriented analyses. We are not aware of any comparative evaluation between such techniques in the literature.

One of the methods that is widely used for enhancing passage retrieval effectiveness is query reformulation. This can be carried out in two main ways: i) query expansion, and ii) query rewriting. Table 2.2 summarizes a number of studies on query expansion and rewriting. Most of these works use linguistic knowledge and suggest higher retrieval effectiveness once query expansion or rewriting

is carried out.

Table 2.2: A summary of the studies on query expansion/rewriting using linguistic or lexical knowledge

Method	Resource or technique	Reference(s)
Query expansion	Natural language generation	(Neumann and Sacaleanu 2004)
Query expansion	WordNet and the Web	(Yang and Chua 2002)
Query expansion	WordNet	(Vicedo 2001)
Query expansion	NOMLEX and Wikipedia	(Katz et al. 2005; Marton and Katz 2006)
Query expansion	WordNet	(Hovy et al. 2000)
Query expansion	WordNet, ASSERT, and PropBank	(Schlaefer et al. 2007)
Query expansion and rewriting	WordNet	TextMap QA system (Hermjakob, Echihabi, and Marcu 2002)
Query expansion	Hyponyms, hypernyms, and synonyms	(Stokes et al. 2007)
Query expansion	Hyponyms, hypernyms, and related concepts	(Zhou et al. 2006)
Query expansion	Syntactic analysis in the LCA method	(Sun, Ong, and Chua 2006)
Query rewriting	Adding/removing keywords	(Moldovan et al. 1999) and (Harabagiu et al. 2000)
Query rewriting	Linguistic structures and dependency trees	(Kaisser 2005; Kaisser and Becker 2004; Kaisser, Scheible, and Webber 2006; Kaisser and Webber 2007)
Query expansion	WordNet and the TREC corpus	(Prager, Chu-Carroll, and Czuba 2001)
Query expansion	Morphological alternations and WordNet	(Harabagiu et al. 2001)
Query expansion	Grammatical information and part-of-speech tags	(Clarke et al. 2000)
Query expansion	Question keywords, question title, answer type, and question topic	(Oh, Myaeng, and Jang 2007)
Query rewriting	Lexicon	(Brill et al. 2001)

The question or query-side linguistic analysis directly linked with a passage-side analysis at the time of retrieval is another online approach to question/query analysis. As one of the avenues for such methods, shallow semantic analysis of the texts of the questions and passages, like that demonstrated in (Harabagiu and Bejan 2006; Hickl et al. 2006; Moreda, Navarro, and Palomar 2005), can be considered. The procedure for answer-bearing sentence retrieval in (Bilotti et al. 2007) takes a similar approach by using the structured semantic representations based on the predicate-argument structures encapsulated in PropBank (Kingsbury and Palmer 2003; Palmer, Gildea, and Kingsbury 2005). Another example is the work in (Ferret et al. 2000) where both questions and document sentences are tagged with their targets. The target may be a PERSON, ORGANIZATION, LOCATION, and so forth which are hierarchised in 17 semantic classes. This information is used in measuring the similarity between the questions and document sentences to retrieve and rank the most related sentences. AnswerFinder, as described in section 2.1.1.1, also performs various forms of question and passage, more specifically sentence, analysis at the lexical, syntactic, and semantic levels to retrieve, score, and rank the sentences that are most likely to contain correct answers to a given question. As mentioned before, at the syntactic level, it uses the grammatical relations

between the parsed questions and sentences and at the semantic level it benefits from the logical forms to identify the semantics that the questions and sentences share. While lexical analysis is the basis for retrieval, both syntactic and semantic analyses are performed to more effectively score and rank single-sentence passages in this work.

2.1.4 Discussion of Key Aspects in Linguistic Passage Retrieval

Retrieval of the most specific passages to queries formed on the basis of natural language questions is dependent on: i) the level of linguistic knowledge used in question and document/passage analysis, ii) the passage boundary detection method, iii) the passage indexing procedure, and iv) the approach of online question and/or passage analysis.

From a linguistic knowledge viewpoint, syntactic information can improve a linguistically unaware process and can be considered in conjunction with the other levels of linguistic knowledge, especially semantics. Morphological analysis of query and passage terms can overcome surface mismatches, but is limited to the different formations of a lemma. However, with different levels of paraphrasing (expressing concepts in different words), it is necessary to exploit semantic knowledge. This knowledge can, for example, capture the similarity between "putting pen to paper" and "writing".

Passage indexing exploiting linguistic knowledge is another aspect that can be considered to more efficiently retrieve passages. However, this higher online efficiency may be blurred by the need for performing the repetitious expensive offline process of indexing. As the size of the collection grows and the amount of linguistic knowledge that needs to be indexed also increases, this converts to a considerable challenge. The situation is more complicated when considering dynamic text corpora such as the Web collection. Reducing the index size and the burden of the indexing task by eliminating linguistic aspects of indexing may help at the expense of some online linguistic analyses on the questions and answer passages. This leads us to the more recent techniques which perform query analysis and in some cases both query and passages analysis. Query analysis is carried out using two main approaches:

- **Query expansion:** to add more contextually or conceptually related terms to the query.
- **Query rewriting:** to reformulate the existing terms in the query so that it can be matched to the other terms in the passages. It is also possible to perform a chain of analyses to semantically link the query and passage terms and concepts.

The first approach increases the recall measure which can reduce the precision value. In the context of QA, a higher precision is more desirable (Pradhan et al. 2002) to the extent that the highly ranked passages should contain answer candidates. It also increases the costs of query evaluation (Kaszkiel, Zobel, and Sacks-Davis 1999) resulting in a more expensive end-to-end QA procedure. Therefore, the query rewriting approach is preferred to query expansion.

To linguistically reformulate a query (and/or passages), there have been many studies as mentioned in section 2.1.3; however, while most of the studies consider the WordNet-based relations - hypernymy, hyponymy, and synonymy - none of them take deep unstructured semantic and scenario-based relations into account. For instance the relations between the pairs "sender-receiver" and "son-mother" cannot be handled by those lexical relations in WordNet or similar resources, although such and similar semantic relations may be required to address the deep paraphrasing instances.

The natural language generation-based approach for answer sentence retrieval and ranking in (Kaisser 2005; Kaisser and Becker 2004; Kaisser, Scheible, and Webber 2006; Kaisser and Webber 2007) uses different rich linguistic patterns and resources that may be able to deal with such complications; however, it only constructs different queries with different syntactic structures that do not afford semantic alternations at any level of lexical or scenario-based relations.

The usage of scenario-based relations in (Hickl et al. 2006) is a big step towards resolving deep semantic relatedness of questions and passages; however, this work still suffers from not having any control on the retrieval part. These relations are employed to affect the scoring and ranking process of retrieved passages. There is no direct evidence in this work that shows how effectively this approach can enhance the answer passage retrieval performance of a QA system.

2.2 Factoid Answer Processing

The task of finding an exact answer to an open-domain factoid question (from parts of specifically related texts) can be more effectively carried out using linguistically aware modules in conjunction with other approaches of pinpointing answer candidates (Bernardi et al. 2003). This is especially the case when there is a clever combination approach which combines both methods of answer processing. This becomes more essential in cases where simple information extraction-based linguistically-impoverished methods and straightforward answer type identification processes cannot solely guide the QA systems to access answer candidates (Narayanan and Harabagiu 2004*b*). Instances of linguistically unaware systems include the data-driven approach for learning the models of answer types, the query content, and answer processing in (Lita and Carbonell 2004), the statistical agent in (Chu-Carroll et al. 2003) for extracting answers using maximum entropy and answer correctness models based on a hidden variable representing the answer type, the data redundancy-based method as the basis for n-gram mining, filtering, and tiling to access the actual answers studied in the AskMSR QA system (Brill, Dumais, and Banko 2002), and the answer redundancy-based approach demonstrated in (Dumais et al. 2002) to extract the most frequent entity as an answer candidate.

The results from the TREC evaluations show that the best-performing system in eight consecutive competitions (Dang, Lin, and Kelly 2006; Voorhees 1999; Voorhees 2000; Voorhees 2001; Voorhees 2002; Voorhees 2003; Voorhees 2004; Voorhees and Dang 2005) has been exploiting linguistic information especially at the level of semantics along with artificial intelligence techniques (logic provers)

to extract answer candidates.

Having considered that linguistic information can play an important role in factoid answer identification, in this section, some of the important relevant aspects of the usage of such information in the task of answer processing will be explained. The major concern of this thesis is on the contributions of a type of linguistic knowledge - namely frame semantics (see section 2.2.3) - in the task of factoid answer processing in conjunction with other types of answer processing such as named entity-based models which extract factual answers as named entities in texts. We believe that frame semantics and its possible ways of contribution to the domain have not been comprehensively studied to date. Therefore, we focus our attention on different linguistic resources that can be used to identify and score factoid answers.

2.2.1 WordNet-Based Processes towards Answer Identification

WordNet is a lexical reference system the design of which is inspired by psycholinguistic theories of human lexical memory (Miller et al. 1990). This domain-independent linguistic system includes all English verbs, nouns, and adjectives organized into synonym sets, also known as *synsets*[3], between which there are different relations. Each of the sets represents an underlying concept and from this point of view this lexical system forms a concept hierarchy with the different abstraction levels of the concepts.

The main organization of WordNet consists of the semantic relations between the synsets. A semantic relation is a relation between meanings where the meanings are expressed in the synsets. The semantic relation set in WordNet contains the relations such as synonymy (between different terms with the same meaning), antonymy (between terms with opposite meanings), hypernymy/hyponymy (between a more general concept and a more specific concept like "motor vehicle" and "car"), and meronymy (between a concept and its containing concept such as "automobile" and "wheel"). There are also morphological relations between different word forms to deal with inflectional morphology in the language.

WordNet, as one of the first linguistic resources available for computational linguistic applications, has been extensively used in the domain of factoid open-domain answer processing. In (Novischi and Moldovan 2006), the WordNet synsets and their relations are exploited for propagating verb arguments along lexical chains[4]. This propagation allows resolution not only of the paraphrasing problem between the verbs that appear in a given question and answer-bearing sentences, but also of the positioning and the role of the arguments which may differ from verb to verb.

The study in (Humphreys et al. 1999) benefits from the WordNet relations between the synsets to

[3] An example of WordNet synsets is the set "girl, miss, young lady, young woman, fille" which includes different words and phrases with the same meaning.

[4] A lexical chain is a set of semantically related lexical items (terms) with WordNet relations between them. For instance, "girl, woman, female, person, organism, living thing, object, entity" shows a lexical chain with the hypernymy relation between the terms.

identify semantically similar and compatible events indicated by the predicates in a given question and its answer sentences (such as "write" and "compose"). They limit the depth of the links in WordNet to be traversed to 3 links. The WordNet-based distance between the events is considered as one of the parameters in measuring the semantic compatibility of the two event classes.

The synonyms and hyponyms for nouns and verbs are extracted from WordNet in (Bos 2006) to form the Discourse Representation Structure (DRS) of both questions and passages along with other linguistic information. The DRSs of questions and passages are compared at the answer processing phase to measure the semantic relatedness of the passage sentences with the questions and to identify a matching score for the answer candidates that can be extracted from the passage sentences. Similarly, in (Monz and Rijke 2001), the synonymy and hyponymy relations from WordNet are exploited to soften the matching process between constituents of the questions and answer passages. The matching score is subsequently used for answer candidate ranking. The number of links traversed in WordNet is taken into consideration as one of the scoring parameters. The hyponymy relations are also used for establishing the lexical relations between entities found in a matching dependency structure and the focus of the question in cases where the question focus is constrained with a main question topic as in the question "What university was Woodrow Wilson president of?". The question focus "university" in this example requires to be related by a hyponym relation to the entities that refer to the name of different universities.

LCC's QA systems have been using WordNet in different ways during the past few years. In (Moldovan et al. 2002), LCC's PowerAnswer system parses the relevant document paragraphs to a given question and transforms them into logic forms. The logic forms together with knowledge axioms extracted from WordNet are fed to a logic prover. The lexical chains that are constructed based on the WordNet relations containing semantically related words improve the answer processing task by linking question keywords with answer concepts. The linkage is resolved by the logic prover articulated as one of LCC's tools which performs the inference rules based on *hyperresolution* and *paramodulation*. The hyperresolution inference excludes a pair of literals if they are the same literals with positive and negative forms in a clause. The result is a newly inferred clause without those literals in any form. The latter - the paramodulation inference - excludes the axioms representing equality in the proof. Both inferences perform multiple steps in one. In (Harabagiu et al. 2003), LCC's QA system combines information extraction techniques, to access named entities, with abductive reasoning[5] on the axioms derived from WordNet and those axioms approximating semantic relations or linguistic pragmatics. Their PowerAnswer-2 QA system (Harabagiu et al. 2005) also exploits the axioms derived from the eXtended WordNet as one of the inputs to the COGEX logical prover (Moldovan et al. 2003*a*) to abductively prove the semantic relatedness of the answer candidates to the question. The eXtended WordNet is an extended version of WordNet semantically

[5] Abductive reasoning is a method of logical inference using which preconditions/explanations of consequences are inferred.

improved by syntactically parsing the glosses - the meanings of terms - and semantically disambiguating the content words (Harabagiu, Miller, and Moldovan 1999). With the semantic clusters in LCC's eXtended WordNet knowledge-base, the accuracy of the lexical chaining module increases in PowerAnswer-3 (Moldovan, Bowden, and Tatu 2006). LCC's eXtended WordNet is a knowledge-base which captures and stores world knowledge in the parsed and sense disambiguated glosses of eXtended WordNet. They add temporal axioms to be fed to the logic prover in the PowerAnswer-3 QA system to more concisely relate answer candidates to a given question with respect to the temporal references in the questions.

The query expansion module in LCC's CHAUCER QA system (Hickl et al. 2006), developed to augment the information retrieval queries for the QA system, also uses the terms from related passages that can be found in the WordNet synsets for a particular question keyword.

The usage of WordNet in the Webclopedia QA system (Hovy, Hermjakob, and Lin 2001), especially to more effectively answer definition questions, is based on extracting the target term definitions from the appropriate WordNet glosses. The answer candidates that the system identifies to a given question affect the identification of the final answer according to their distance to the definitions from the WordNet glosses. In (Na et al. 2002), the answer processing task is generally encountered as a named entity-based approach that identifies the entities in the passages as candidate answers to a given question based on taxonomic relations in WordNet between the entities and the answer type of the questions. In cases where the answer type is a hypernym of the entity type, the entity is considered as an answer candidate. If the entity type is a hypernym of the answer type the answer identification requires more contextual analysis of the entity. The system considers the relations as one of the parameters for answer scoring and ranking. The logic forms in (Rus 2002) based on the eXtended WordNet are also demonstrated to have application in boosting the performance of answer extraction and ranking. The idea is based on using hyper-inference and axiom-inference to provide explanations on the answer candidates extracted using the WordNet lexical chains between the pairs of concepts in a given question and its candidate answer paragraphs.

ExtrAns (Molla et al. 2003) is another QA system that benefits from WordNet relations in a logical inference mechanism to identify answer candidates. The main relations used in ExtrAns are the hyponymy and synonymy relations. In the synonym stage, the word senses are handled by randomly assigning the synsets. They argue that in the context of technical texts, as the main domain of their QA system, the word senses have minimal impact on the task of QA as the words in such a domain have limited ambiguity. The answer processing module in ExtrAns, using the inference steps at the synonym stage and hyponym stage, performs different actions to more effectively pinpoint answer candidates in the answer corpus. It replaces the terms in the logical form of the queries with their synonyms in the synonym stage and adds the hyponyms as disjunctions to the logical form in the hyponym stage. There are other alternative inferences in the system such as distributivity of conjunctions, approximate matching, and keyword matching.

WordNet and eXtended WordNet are also used for verb expansion in (Sun et al. 2005). They find similar verbs using the two linguistic resources when matching the shallow semantic predicate-argument structures based on the verbs in a given question and corresponding answer passages. The result of this verb expansion is that their system can deal with the situations where the same events are expressed using different verbs.

The answer justification method to filter out semantically erroneous answers to a given question is another approach used in (Harabagiu et al. 2000) which extracts the world knowledge axioms from the gloss definitions of WordNet. They show that the option of semantically justifying the list of answer candidates with the world knowledge axioms, textual answer facts axioms, and co-reference axioms can improve the effectiveness of correct answer detection by a linguistically-aware answer processing module.

Overall, the WordNet glosses and synsets have been used mainly for two purposes: i) to find lexical semantic relations between different concepts and keywords in the questions and answer-containing text snippets, and ii) to encapsulate world knowledge in the process of answer identification and justification. While the first direction improves the systematic understanding of the texts in the presence of different types of surface mismatches, the second direction elevates the level of confidence of the systems especially with answer redundancy being a major challenge in answer identification, justification, and ranking. Both these directions have resulted in improvements in the task of factoid answer processing over recent years.

It should be noted that the contributions of WordNet-based approaches focused on the other subtasks of QA - specially question analysis - are neglected in this section as they are not the focus of this thesis. Some of the other different aspects of the usage of WordNet in the domain of QA have been studied and may be found in (Paşca and Harabagiu 2001).

2.2.2 PropBank in Answer Processing

Proposition Bank, known as PropBank (Kingsbury and Palmer 2003; Palmer, Gildea, and Kingsbury 2005), is a project at Penn (University of Pennsylvania), which aims at adding semantic annotations to the syntactic structures already present in the Penn TreeBank (Marcus et al. 1994; Marcus, Santorini, and Marcinkiewics 1993). In this project, verb predicates are annotated with their arguments leaving the other part-of-speech predicates aside. This information is stored in semantic frames which encapsulate the predicate-argument structure of the verbs.

The verbs of sentences typically present the events that happen with regard to the different participant roles in the events (Kingsbury, Palmer, and Marcus 2002; Surdeanu et al. 2003). In the sentence "The futures halt was assailed by Big Board floor traders.", for example, "the futures halt" plays the role of the thing or person assailed and "Big Board floor traders" is realized in the role of the assailer, while the main action in the sentence is "assailing". The associated predicate-argument structure for this sentence would be *assail(Big Board floor traders, the futures halt)* where

the different roles are considered as the arguments of the verb predicate "assail". As another instance, the verb "hit" with the sense "strike" needs at least three roles to be present in a grammatical and meaningful sentence: i) hitter, ii) thing that is hit, and iii) instrument with which the action of hitting is performed. As a result, there can be a structure as below:

Hit (sense: strike)
Arg0: hitter
Arg1: thing hit
Arg2: instrument

The arguments in PropBank are numbered as Arg0, Arg1, and so forth depending on the valency of the verb under consideration. This model of argument labeling has been chosen in order to make the annotations easily mappable onto the labels used in most modern theories of argument structure (Kingsbury and Palmer 2003). Therefore, the sentence "Alice hit the dog with a strap." can be annotated in this sense like as "$^{Arg0:}$ Alice $^{Predicate:}$ hit $^{Arg1:}$ the dog with $^{Arg2:}$ a strap". Another example can be the verb "edge" in the sense of "move slightly" which has the arguments as below:

Edge (sense: move slightly)
Arg0: causer of motion
Arg1: thing in motion
Arg2: distance moved
Arg3: start point
Arg4: end point
Arg5: direction

This is a more complicated verb which may require six arguments to be present in a structurally complete sentence. The annotated form of the example sentence "Her car edged the fence towards our house." with respect to the predicate "edge" is "$^{Arg0:}$ Her car $^{Predicate:}$ edged $^{Arg1:}$ the fence $^{Arg5:}$ towards our house.". In many cases there are one or more arguments missing in a sentence, without causing the sentence to be incorrect or semantically incomplete. As an example, the sentence "Her car edged the fence." is still grammatical, although it misses some arguments of the main predicate "edge".

Knowing that verbs may have diverse senses in different contexts, it is very important to first disambiguate the sense of the verb in most related applications. The disambiguation is performed on the basis of differing argument structures which occur in the predicate-bearing sentence. The number of arguments and the semantic ground of them are the main criteria for sense disambiguation on the basis of this structure with a greater emphasis on the number of arguments (Kingsbury and Palmer 2003).

The predicate-argument structure encapsulated in PropBank relates different verbs to different types of nouns as the units of the event that the verbs cover (Kawahara, Kaji, and Kurohashi 2002). Such semantic information focused on the verbs in the questions and answer passages can be useful for more precisely answering factoid questions. The general notation of predicate-argument structure is used in a QA system developed in (Kawahara, Kaji, and Kurohashi 2002) to extract factoid answers.

Answer identification is attained by matching the structures. The matching process looks for the case components that share the same instances. Structures with at least one same case component are classed as match structures. The case components corresponding to interrogative question stems are instantiated with the values of the corresponding case component in answer passages. This type of semantic alignment results in a more relaxed procedure which does not necessitate all of semantic roles to match in question and passage structures. As one of the frame semantic-based answer processing methods, we implement a more relaxed strategy of semantic alignment in Chapter 6 which shows a higher performance compared to a complete semantic role matching procedure.

One of the early attempts to utilize the semantic layer of information that can be contributed to texts by using the predicate-argument structure in PropBank is made in (Narayanan and Harabagiu 2004*a*). The semantic role of the answer candidates is identified according to the role of the question stem in a given question. Their work progresses in this direction in (Narayanan and Harabagiu 2004*b*). They consider other steps of deep semantic analysis of the questions and answer passages such as i) a more articulated identification of the topic model of the scenarios in which the question is being asked, and ii) the further event modelling of the actions in complex scenarios to provide a scalable model for conducting reasoning and inference processes in QA in the presence of inter-related complicated scenarios. The benefit of all these inferences and relations is that they assist the system to recognize the situations where a passage does not contain the exact correct answer to a given question, although it includes the question keywords. For instance, the sentence "Ruby killed Oswald." does not include the answer to the question "Who did Oswald kill?" (Bilotti et al. 2007) although it contains all of the question keywords. These types of relational constraints are considered in the structured semantic representations used in (Narayanan and Harabagiu 2004*b*) along with a probabilistic inference technique.

The predicate-argument structures in PropBank are also used for semantic matching of the answer passages with WordNet-based verb expansion in (Sun et al. 2005). Answer passages are ranked according to the argument and verb similarities in questions and passages and the answer extractor module retrieves top ranked entities and arguments (in cases that the answer type is not a named entity class) as answer candidates. The Jaccard coefficient is used to measure the argument similarity between two semantic predicate-argument structures.

A different approach in CHAUCER (Hickl et al. 2006) uses a PropBank-based semantic parser to generate natural language predictive questions on the basis of each predicate found in the top-ranked passages per question. A set of heuristics is used to identify one of the arguments of each predicate as the answer of a generated factoid question and then, the argument is mapped to one of the *wh*-phrases (such as *who*, *when*, and *where*). The predictive questions, as question-answer pairs, are used as one of the answer processing techniques in CHAUCER to generate answer candidates based on the similarity metrics between the original question being answered and the list of predictive questions.

Two different methods are exploited for factoid answer processing based on the linguistic structures by Kaisser, Scheible, and Webber (2006) and Kaisser and Webber (2007). In their first method, they generate natural language sentences based on PropBank and other resources - namely FrameNet and VerbNet (Schuler 2005) - which have some known components and at least one unknown component as the answer segment. They make use of a semantic role labeling technique to annotate the question with its predicates and arguments. Then the sentences, as the answer templates, are generated from the abstract semantic frames in the resources. Queries based on the generated templates are sent to the sentence retrieval module and the related sentences with the same semantic structure are then annotated accordingly. The answer candidates are extracted as the fillers of the same semantic roles of the vacant answer segments in the questions. Their second method of answer processing considers the dependency structures in the example annotated sentences of PropBank (and FrameNet) and the list of retrieved related sentences to the query formed on the basis of the question keywords. Once the list of related sentences has been retrieved, there is a list of criteria on which to check the dependency structure of them. This method combines the linguistic knowledge at both levels of semantics (by semantic roles) and syntax (by dependency structures). According to the check list of the dependency structures, the answer candidate-containing sentences are scored and ranked. The final answers are extracted from the top-ranked sentences.

The ASSERT shallow semantic parser (Pradhan et al. 2004) is used in (Schlaefer et al. 2007) to construct a PropBank-based semantic representation of fact-seeking questions and extract answer candidates from answer sentences which have similar representations. The extraction of related NEs from all of the arguments of answer predicates in cases where EATs are known makes their approach more robust against low accuracy semantic role labeling.

From the studies conducted using the predicate-argument structure in PropBank for the answer processing task, it can be seen that the PropBank semantic frames on verbs can offer great opportunities for identifying factoid answers in an effective manner. This is achieved in two ways: i) by the usage of semantic knowledge that can be inferred and contributed to texts as a semantic layer for better text understanding and event handling, and ii) by the structure of the sentences in verb semantic frames that can be used for the syntactic analysis of related text snippets to a given question.

2.2.3 FrameNet-Based Techniques to Answer Detection

Frame semantics, basically developed from Charles Fillmore's Case Structure Grammar (Cook 1989; Fillmore 1968), emphasizes the continuities between language and human experience (Fillmore 1976; Lowe, Baker, and Fillmore 1997; Petruck 1996). The main idea behind frame semantics is that the meaning of a single word is dependent on the essential knowledge related to that word. With such an understanding of frame semantics, the required knowledge about each single word is stored in a semantic frame. In order to encapsulate frame semantics in such frames, the FrameNet project

(Baker, Fillmore, and Lowe 1998) has been developing a network of inter-related frames which is a lexical resource for English currently used in many natural language applications.

The main entity in FrameNet is the *semantic frame* which develops a kind of semantic normalization over concepts semantically related to each other. The semantic relation between concepts in a frame is realized with regard to the scenario of a real situation which may happen and cover the participant concepts rather than synonymy or other such relations like hypernymy and antonymy. In this regard, the frames encode the base definitions necessary to understand the semantics and the scene of each member term. In other words, real-world knowledge about real scenarios and their related properties are encoded in the frames (Lowe, Baker, and Fillmore 1997). Each frame contains some *frame elements* (FEs) as representatives of different semantic roles regarding a target predicate inside the frame. The semantic roles are common properties among all of the terms that are inherited from a frame. This ensures a suitable inclusion over the English terms which either have similar meanings or share the context and/or the scenario in which they can occur in the sentences of the language.

A limited set of frame-to-frame relations has been defined in FrameNet which connects frames to constitute a network of concepts and their semantic pictures (Ruppenhofer et al. 2005). Such relations have been used for event structure identification and inference on complicated story scrutinizing in applications like QA (Kaisser, Scheible, and Webber 2006; Kaisser and Webber 2007; Narayanan and Harabagiu 2004*a*; Narayanan and Harabagiu 2004*b*) which will be discussed later in this section.

In addition to the associations between the FEs across the frames necessitated by the frame-to-frame relations, there are other relations between the FEs within a frame. The two major within frame FE relations are the *requires* and *excludes* relations which emphasize the philosophy of existence of the participant roles in the scenarios covered by the definitions of the frames.

The FrameNet database, as mentioned in (Baker, Fillmore, and Lowe 1998), contains three main components:

- **Lexicon:** that contains entries which are composed of i) conventional dictionary type data, ii) formulas for capturing the morpho-syntatic ways in which elements of the semantic frame can be realized, iii) the links to semantically annotated example sentences, and iv) the links to the frame database and other machine-readable resources,
- **Frame database:** which contains all of the frames, their FEs, and corresponding definitions and semantic types, and
- **Annotated example sentences:** that collects all of the exemplified sentence annotations with the FrameNet frames and FEs to provide empirical support for lexicographic analysis provided in FrameNet.

Table 2.3 shows an example frame "Manufacturing" with its definition, core FEs, and predicates (lexical units). The main semantic roles that are necessary for the scenario to be complete are those known as the core FEs "factory", "manufacturer", and "product". A number of lexical units with

different parts-of-speech (such as noun, verb, and adjective) inherited from this frame are shown in the last row.

Table 2.3: An example FrameNet frame

	Frame: Manufacturing	
Definition	A **Manufacturer** *produces* a **Product** from **Resource** for commercial purposes.	
	FACTORY	Those machines were *manufactured* **in the Miami plant**.
FEs	MANUFACTURER	**General Electric** *produces* electric appliances.
	PRODUCT	The company *manufactured* **many T-shirts**.
LUs	fabricate.v, fabrication.n, industrial.a, make.v, maker.n, ..., production.n	

The scenario-based relations between the lexical units captured in the frames is one of the main differences between FrameNet and other linguistic resources such as PropBank and WordNet. This can be better realized in the frame "Kinship" where some of the lexical units are "father", "kid", "son", "mother", "aunt", and "uncle". The relationship between these terms is none of the synonymy, hypernymy, hyponymy, or antonymy relationships; instead, they are related to each other and covered by a single frame only because they participate in contextually similar events. These events are represented/modeled by the main event expressed in the frame "Kinship".

Since terms with different parts-of-speech can participate in a certain event or state, FrameNet frames may cover predicates of any part-of-speech. For instance, the frame "Taking_time" covers a list of adjective ("fast", "quick", "rapid", "speedy", "swift"), preposition ("in"), adverb ("slowly"), and verb ("take") predicates.

Another main difference from PropBank is the different parts-of-speech predicates that are inherited from the semantic frames in FrameNet as opposed to the verb-only semantic frames in PropBank. This allows the addition of semantic roles to the arguments of the different part-of-speech predicates in free texts with those semantic roles in the FrameNet frames. NomBank (Meyers et al. 2004*b*), however, extends the scope of PropBank by providing the same type of semantic frames for the noun predicates in the PropBank corpus - the Wall Street Journal Corpus of the Penn TreeBank. There are other part-of-speech predicates such as adverbs and adjectives which are not yet semantically structured in this corpus to the extent available in FrameNet.

The other difference between FrameNet and PropBank is in the representation of the frames. The semantic frames in PropBank are predicate-oriented (there is one frame per predicate) and in FrameNet there is a generalization over a number of predicates which share the same semantic structure. This leads to different argument labels in PropBank for the same roles (such as "buyer") and semantically similar predicates like "sell" and "buy" (Fliedner 2004). In FrameNet, however, there is no such diversity as the semantic roles are known through the names of the FEs.

With such benefits of frame semantics encapsulated in FrameNet, it has been exploited in few

QA studies[6] and we could not find any work that describes its full contribution to QA in different directions.

Of the first studies which analysed the usage of frame semantics in the answer processing task is the work by Narayanan and Harabagiu (2004*a*) and Narayanan and Harabagiu (2004*b*). They utilize the semantic structures in PropBank and FrameNet to more effectively identify the question model and resolve complex event structures in answer passages. As mentioned in section 2.2.2, their inference mechanism can resolve scenario-based relations between the different events realized by the predicates in free texts to fully semantically identify answer candidates. They use the PropBank predicate-argument structure where the incomplete lexical coverage of FrameNet does not allow for frame semantically annotating questions and answer passages. Although this study does not cover many other aspects related to FrameNet-based answer processing, it clearly demonstrates the capability of these linguistic resources (PropBank and FrameNet) in elevating state-of-the-art QA systems when combined with a sophisticated event representation mechanism and an appropriate inference methodology. In (Fliedner 2004), there is a study on automatically deriving FrameNet representations from the free text of the corpus documents and questions to be useful for effectively pinpointing the answer sentences and identifying the candidate answers in a QA system by matching the FrameNet structures in both questions and answer sentences. They suggest the support of frame granularities by considering hypernym and hyponym search in frame matching. The introduction of underspecified *pseudo-frames* is suggested to be required for coping with the words and concepts that are not covered by FrameNet. They do not, however, conduct any real QA experiments using the FrameNet structures.

The syntactico-semantic analysis in QuALiM (Kaisser 2005) with semantic roles of the FrameNet frames assists the system to find pieces of evidence from the example annotated sentences in the frame to identify the type of syntactic relation between the head predicate of an answer sentence and an answer candidate. There is a blurred process of frame matching and semantic alignment strategy taken into consideration in this approach as the answer sentences are retrieved according to the extracted queries by analysing the example sentences in the frames from which the head predicates inherit. An answer candidate is identified as a segment at the specific syntactic relation to the predicate in the answer sentences. This methodology is further elaborated by using other linguistic resources - PropBank and VerbNet - and more comprehensive semantic methods of answer processing (Kaisser, Scheible, and Webber 2006; Kaisser and Webber 2007). Their methods have been explained in section 2.2.2. More specifically, in the context of FrameNet, they argue about the usage of the inter-frame relations to overcome a wider range of paraphrasing challenges and frame granularity, although it is not clear how many levels of frame links they may follow to generate additional answer templates.

LCC's CHAUCER QA system (Hickl et al. 2006) uses a frame alignment and FE matching

[6] One such study can be found in our preliminary work published in (Ofoghi, Yearwood, and Ghosh 2006*a*).

strategy based on FrameNet as one of its different answer processing techniques. They exploit LCC's FrameNet parser to annotate the text of the questions and passages with the FrameNet frames and FEs. The retrieval of the answer passages are biased towards the passages with similar frame distributions and the parser's confidence in assigning the frames to the passages. Identification of the answer candidates in this technique is based on instantiating the vacant FE in the question with the string value of the corresponding FE in the answer passages. All of the answers from the different answer processing techniques in CHAUCER are re-ranked using a Maximum Entropy- based algorithm which takes into account different features of the answers from the different techniques.

One of the recent efforts in this direction is the study conducted in (Shen and Lapata 2007) which formulates the usage of semantic role labeling via bipartite graph optimization and matching for answer processing using FrameNet frames and FEs. Their approach benefits from a soft semantic role labeling and an optimization method to overcome the problem of multiple- (and/or no-) labels for the semantic roles. The soft labeling outputs in the form of graphs are consequently used for scoring the answer candidates. Their answer candidate identification process does not perform any FE matching; instead, it extracts the noun phrases, as answer candidates, with the same named entity type as the EAT of the question. The experiments by Shen and Lapata show the improvement over non-FrameNet-based and non-semantic-role-based answer processing techniques. There is, however, no individual evaluation on each separate task of class identification and role labeling performed in this study.

In the area of domain-specific QA, the biological version of FrameNet, BioFrameNet (Dolbey, Ellsworth, and Scheffczyk 2006), has been suggested to be a useful resource for leveraging answer processing effectiveness, although it has not been directly exploited in this direction to date. BioFrameNet extends FrameNet with domain-specific semantic relations and is linked to domain ontologies such as the Gene Ontology. With such characteristics, BioFrameNet is expected to be beneficial in the context of biological QA by conducting reasoning processes such as what is demonstrated in (Scheffczyk, Baker, and Narayanan 2006).

In general, FrameNet has been used for answer processing in three main ways: i) to extract the underlying semantic (and syntactic) information about predicates to be used in natural language generation-based approaches for more precisely retrieving answers, ii) for semantic alignment between questions and answer sentences to identify specific passages and extract answer candidates, and iii) for event modelling and scenario-based analysis of answer sentences.

With this said, there has not been sufficient work on uncovering many related aspects of using FrameNet for factoid QA systems. This includes the level of text annotation with FrameNet elements, different methods of answer processing, the linguistic coverage of FrameNet, and the technique of fusing FrameNet-based answer processing models with other answer processing models (such as entity-based models). We will discuss the aspects which have led us to this study in section 2.2.6.

2.2.4 Other Linguistic Resources

To develop linguistically aware QA modules, there are other linguistic resources that have been used. In many cases they have been exploited as a complementary resource in conjunction with linguistic resources already mentioned.

VerbNet (Schuler 2005) is one such resource which is a verb lexicon for English that extends Levin's verb classes (Levin 1993). It is a domain-independent wide-coverage resource linked to some other resources such as WordNet, FrameNet, and Xtag[7]. Both types of syntactic and semantic information are encapsulated in the VerbNet classes for each sense of a given verb. The thematic roles of each verb, the selectional restrictions on the arguments of the verbs, and the syntactic description and semantic predicates are represented by classes in VerbNet each of which covers a group of verbs. The verb grouping procedure is based on syntactic criteria as in Levin's verb classes where the shared syntactic information reflects the same semantic properties.

The verb arguments in VerbNet are assigned to thematic roles within the classes where a thematic role can be an ACTOR, AGENT, ASSET, or ATTRIBUTE. VerbNet is used in the context of QA in (Novischi and Moldovan 2006) which makes use of the syntactic patterns of the verbs and to use this structure as strong evidence for extracting an answer candidate. The syntactic patterns contain tokens such as thematic roles, the verb itself, adjectives, adverbs, prepositions, and plain words. The propagation of verb arguments along the syntactic patterns and the WordNet relations between different verbs, as mentioned in previous sections, allows for matching differently expressed concepts in a given question and its answer sentences.

The work in (Amoia and Gardent 2005) to recognize difference verbalizations of the same concepts and overcome the surface paraphrasing problems using a shallow parser (called XIP) can also be considered in the area of answer processing. The XIP shallow parser is powered by the linguistic information from VerbNet to deal with alternation paraphrases. Of the other QA studies that use VerbNet, the works by Kaisser, Scheible, and Webber (2006) and Kaisser and Webber (2007) can be considered where VerbNet is exploited for answer processing along with other linguistic resources, FrameNet and PropBank (see section 2.2.2).

NomBank (Meyers et al. 2004*b*) is another linguistic resource which has been used in some QA studies. It follows the PropBank annotations by encapsulating nominalised predicates in the same way that PropBank covers verb predicates. This allows for constructing predicate-argument structures based on noun or nominalised predicates. We have not found any study on factoid answer processing techniques that benefit from the NomBank frame files. However, they are used for question and document processing in (Hickl et al. 2006) to identify semantic dependencies between sentence constituents. NomBank frames are also used for event modelling in an information extraction-based QA process in (Schiffman et al. 2007) in conjunction with PropBank. They are used for question analysis to guide temporal inference and answer complex time-based questions in (Harabagiu and

[7] http://www.cis.upenn.edu/∼xtag/

Bejan 2006) again along with PropBank.

The NOMinalization LEXicon - NOMLEX - (Meyers et al. 1998) is a lexicon of English nominalizations developed in the Proteus Project at New York University. It contains allowed complements of nominalizations - deverbals - and also relates these nominal complements to the predicate-argument structure of the corresponding verb. The advantage of using this lexical resource is to identify the conceptual similarity of a noun phrase such as "Rome's destruction of Carthage" and the sentence "Rome destroy(ed) Carthage" (Macleod et al. 1998). Therefore, NOMLEX encapsulates information about the verb arguments that can be found in their deverbal or nominalised forms. NOMLEX includes such information on about 1000 English verbs and has been extended to NOMLEX-PLUS (Meyers et al. 2004*a*) with respect to other parts-of-speech-based nominalizations and covers about 5000 deverbal as well as de-adjectival and de-adverbial nouns.

The La Sapienza QA system (Bos 2006) benefits from the background knowledge that can be inferred from WordNet and NOMLEX in the question processing task and answer processing phase. The knowledge is used in matching the passage sentences with a given question to identify a matching score for any answer candidate extractable from the passages. A relatively similar approach is taken in Tequesta (Monz and Rijke 2001) to extract the verb group and noun group dependency structures both in documents and questions. The dependency structures, extracted from NOMLEX in the nominalization cases, are then used for matching the questions and document sentences in the answer extraction and scoring task. To overcome the paraphrasing instances formed and caused by nominalizations, the NOMLEX lexicon is used in (Rinaldi et al. 2003). Different types of paraphrasing are handled in this work that can elevate the performance of the ExtrAns QA system (Molla et al. 2003). The paraphrases are dealt with to transform both documents and questions to the minimal logical forms and extract answer candidates from logically related answer sentences (Molla et al. 2000).

2.2.5 Other Linguistic Structures

One of the other approaches to performing linguistic analysis in QA, and more specifically answer processing, is to translate the information in the questions and answer passages to an intermediate structure without using any specific linguistic resource.

Ternary expressions (TEs) introduced in (Katz 1990) are such intermediate online transformations over texts which relate the subjects and objects of the sentences via different relations in the format of the triples such as <SUBJECT relation OBJECT>. For example, the sentence "Bill surprised Hillary with his answer" can be syntactically parsed to the two TEs "<<Bill surprise Hillary> with answer>" and "<answer related-to Bill>". The matching process between the TEs of a given question and those of passages or documents, to identify an answer candidate, may fail because of the existence of the different surface syntactic forms. For instance, the sentence above can be alternatively written as "Bill's answer surprised Hillary". In this case, the two TEs are "<answer

surprise Hillary>" and "<answer related-to Bill>". To deal with such paraphrasing challenges, the START QA system (Katz 1997) introduces S-rules which make explicit the relationship between the alternate realizations of the arguments of different verbs. The S-rule for the example predicate "surprise" is formulated as:

surprise S-rule
if <<SUBJECT surprise OBJECT1> with OBJECT2>
then <OBJECT2 surprise OBJECT1>

which encapsulates the different syntactical forms of the realization of a sentence with the predicate "surprise". To obtain more generalized rules which apply for a group of semantically similar verbs, the S-rules are generalized into classes of verbs according to the semantics that they share. A generalized form of the "surprise" S-rule is:

property-factoring S-rule
if <<SUBJECT verb OBJECT1> with OBJECT2>
then <OBJECT2 verb OBJECT1>
provided verb $\in$ emotional-reaction class

in which the clause *provided* imposes the condition in accordance with which a verb can be treated using the "property-factoring" S-rule. In this example, the verbs are required to reflect emotional reactions in order to be treated using the generalized S-rule "property-factoring".

The START (Katz 1997) and Sapere (Katz and Lin 2003) QA systems benefit from the TEs to retrieve the most specific passages to the questions and identify answer candidates by matching the transformations.

Logical form transformation is another avenue to combine linguistic knowledge and logical axioms and proof in the domain of QA. Different approaches in this context are considered to more effectively extract and/or justify answer candidates (Elworthy 2000; Harabagiu et al. 2005; Harabagiu et al. 2003; Harabagiu, Paşca, and Maiorano 2000; Hickl et al. 2006; Moldovan, Bowden, and Tatu 2006; Moldovan et al. 2002; Moldovan and Rus 2001; Molla et al. 2003). The main common procedure among most of the logic-based systems is that they translate the questions and answer passages to the logical forms that can be further analysed to pinpoint the answer candidates or justify and validate the answers extracted using any other techniques of answer processing. However, each system employs a different form of the logical axioms constructed on the questions and passages.

The Flat Logical Form is one of the recent types of the logical transformation which converts the traditional nested logical forms to the flat forms by reifying all the predicate expressions and using the reified entities to refer to these expressions (Molla 2001). One of the advantages of this logical form is its capability of expressing partial information in the text by partial logical forms which can be useful in answer processing. The AnswerFinder QA system (Molla 2003; Molla and Gardiner 2004; Molla and Gardiner 2005; Molla, Zaanen, and Pizzato 2006), uses the flat logical forms in both answer sentence ranking and answer processing phases. This system also employs another specific type of logical transformation called Logical Graphs (Molla and Gardiner 2005). Logical graphs, also used in (Zaanen and Molla 2007) in a multi-lingual QA setting, are automatically constructed

on the basis of the flat logical forms and represent a simplification of this type of logical forms. A logical graph is a directed bipartite graph containing two types of nodes:

- **Concepts:** which refer to the different objects, events, or states, and
- **Relations:** which link the concepts at a level close to the syntactic level. The relations can be grammatical roles or prepositions all labelled by numbers.

Figure 2.2 shows a simple example sentence and its logical graph constructed from the flat logical form. The node with the label "1" refers to the grammatical role "subject" for the event "go" realized in the node "Ellen".

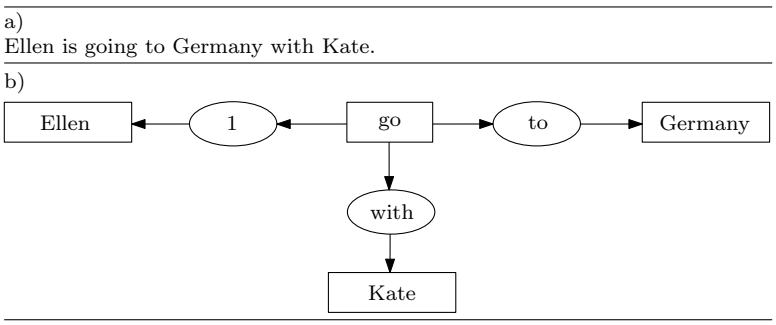

Figure 2.2: Logical transformation in AnswerFinder; a) input sentence, and b) logical graph

The answer processing task, based on the logical graphs, requires the learning of Logical Graph Rules (Molla and Zaanen 2005). These rules contain information on the *graph overlaps* between questions and answer candidate sentences, the *path* from the overlaps to the actual answers in the answer sentences, and the graphs representing the answers. The graph overlap is the graph consisting of the common concepts and relations between the two logical graph forms. A path between two sub-graphs is a sub-graph that connects the two sub-graphs. With a set of logical graph rules learnt from a training set of questions and answer-containing sentences, the answer processing procedure can be approached by testing all the learnt rules to decide whether a sentence answers a given question. This requires questions and answer sentences be transformed into the logical graph forms. The details of this method can be found in (Molla and Gardiner 2005).

The shallow semantic representation of the texts with generic Thematic or Semantic Roles is another approach to more semantically identify and score answer candidates. Such thematic roles, being more generic than the semantic roles in FrameNet and PropBank, can improve the coverage of the representation and matching over the surface structures of the texts. They can also be useful to the extent that the answer boundary detection is skipped by relying on the strings - arguments - that fill the thematic roles. A set of such thematic roles is described and exploited in (Pradhan et al. 2002) to enhance answer identification. Their set of thematic roles includes AGENT, PATIENT, MANNER, DEGREE, CAUSE, RESULT, LOCATION, TEMPORAL, FORCE, GOAL,

PATH, PERCEPT, PROPOSITION, SOURCE, STATE, and TOPIC that can be assigned to the arguments of the predicates using a statistical classifier trained on the FrameNet database. The answer processing task is performed by finding the filler of the thematic role for which the question asks, in case the answer type is a known thematic role.

In (Chai and Jin 2004), linguistic knowledge at the discourse level is articulated to answer context questions. The context questions are those submitted by users in an online interactive QA environment where the questions are contextually related to each other around a target topic. The discourse analysis of the texts of the questions is based on the semantic roles that are mapped to the discourse roles. There is no specific set of semantic roles considered in this work, although possible sets are mentioned to be the FrameNet and PropBank semantic role sets. With the semantic-rich discourse modelling and representation in directed acyclic graphs, the work is suggested to be fruitful in different aspects of QA such as query expansion, inference, summarization, and collaborative QA.

2.2.6 Discussion of Key Aspects in Linguistic Factoid Answer Processing

Having reviewed a number of linguistic resources and their utilization in answer processing, we have found that using different linguistic resources in answer candidate identification and ranking is dependent on the following factors:

- The expressiveness of the resource
- Coverage on linguistic items
- The level of representation of linguistic information of texts
- The effectiveness of the answer processing method
- The fusion of different answer processing methods which benefit from different resources

The resources reviewed above show different levels of expressiveness. Although WordNet and eXtended WordNet provide a good taxonomy of semantic relations to encapsulate world knowledge of lexical items, they have not been, however, designed to reveal the predicate-argument information about any part-of-speech predicates which can help automated text understanding. PropBank, instead, has such information without any taxonomic information on the semantic relations such as synonyms, hyponyms, and hypernyms. It also lacks the information about non-verbal predicates which are partly covered in NomBank that provides predicate-argument information on noun predicates. In none of these resources can a classification of the verbs, like that contained in VerbNet, be found. The deverbals (verb-based nominalizations) are expressed in NOMLEX which has been extended to cover de-adjectival and de-adverbial nouns in NOMLEX-PLUS. Finally, the scenario-based relations between different part-of-speech predicates, as well as inter-scenario relations, are only expressed in FrameNet. From this viewpoint, each resource contributes a unique type of linguistic information towards the text understanding process necessary for QA.

The other linguistic structures - namely logical transformations, ternary expressions, and generic thematic roles - express relatively dissimilar information to that encapsulated in the different linguistic resources, except for the generic thematic roles which have similarities with the semantic roles in FrameNet and PropBank. These structures have no specific bindings to any of the linguistic resources discussed above. Logical graphs capture useful linguistic knowledge, but still suffer from the fact that they are dependent on the syntax of the texts to the extent that the implicit relations cannot be derived where there is no explicit reference to them. For example in the sentence "Kate, 32, is Jack's mother" the "age" relation cannot be expressed until the sentence is converted to "Kate is 32 and is Jack's mother". Even with this translation performed, if a question asks "How old is Kate?" the structure will not be able to distinguish the number "32" as a reference to Kate's age. The ternary expressions will have difficulty coping with such situations as well. FrameNet-based parsing and thematic role-oriented analyses once completed can effectively handle such situations, however.

From a coverage perspective, the ongoing development of the resources promises more chances of having a greater number of lexical items covered by each resource, although for the time being, the wide coverage of WordNet takes this resource to the top of the list. NOMLEX has had a progress to NOMLEX-PLUS (from 1000 nominalizations to 5000) and VerbNet also has a reasonable coverage with 237 hierarchically organized verb classes containing some 5000 verbs. However, compared to WordNet, VerbNet only covers 19.2% of the verb senses in WordNet. PropBank, containing about 4000 frames, develops the coverage over verbal predicates seemingly better than the verb coverage in VerbNet (Pazienza, Pennacchiotti, and Zanzotto 2006). In the literature, there is no explicit report on exact FrameNet coverage over different predicates; however, it has had an increasing number of lexical items covered in its different releases. Generally, FrameNet provides a deep and rich set of semantic structures at the expense of lexical coverage.

Another important aspect is the level of representation of linguistic information of texts by each resource. At this stage, this is imparted on the text parsing level rather than the richness of the resources. Many parsers have been developed after studies on the different characteristics and challenges in adding required linguistic knowledge to texts. The shallow semantic parsing issues related to FrameNet elements more formally starts in (Gildea and Jurafsky 2002) and continues in other works (Erk 2006; Erk and Pado 2005; Erk and Pado 2006; Frank 2004; Giuglea and Moschitti 2006; Honnibal and Hawker 2005; Litkowski 2004; Shi and Mihalcea 2004; Thompson, Levy, and Manning 2003). Previous efforts (before Gildea and Jurafsky's study) were more focused on grammars and data-driven approaches. The task of making FrameNet-based parsing automated requires careful analyses with respect to the sub-tasks of frame evocation and FE assignment. While the first task is considered to be a predicate sense disambiguation problem and finding the right semantic class of the predicate, the second one - the FE assignment task - is a semantic role labeling challenge that necessitates the usage of different syntactical and semantic features of predicate arguments to detect

the argument boundaries and assign each argument to its corresponding FE.

In the case of PropBank, the ASSERT shallow semantic parser (Pradhan et al. 2004) exploits the Support Vector Machines (SVM) classifiers (Cortes and Vapnik 1995) to assign semantic roles to the different parts of texts (the arguments of the predicates). Other PropBank-based studies on shallow semantic parsing can be found in (Chen and Rambow 2003; Gildea and Hockenmaier 2003; Gildea and Palmer 2002; Moschitti et al. 2005; Nielsen and Pradhan 2004; Paek et al. 2006; Pradhan et al. 2003; Surdeanu et al. 2003; Xue and Palmer 2004). The task of shallow semantic parsing with PropBank semantic frames is relatively similar to that of FrameNet as the parser should find the correct semantic frame of the verb predicate and perform a consequent semantic role labeling task.

For some resources, like WordNet, it is just a problem of term lookup and relation processing between the synsets. For generic thematic roles similar parsers to the FrameNet-based and PropBank-based parsers can be adopted as in (Pradhan et al. 2002).

Generally, the resources with more semantic information require more sophisticated text parsing processes to take good advantage of them. For such resources, the accuracy of the parsing task is crucial as it can affect the overall performance of the natural language applications which exploit these resources. Therefore, it is important to carry out experiments and measure the effect of the different levels of shallow semantic parsing on such natural language applications.

There have been many different approaches and methods exploiting the different linguistic resources and structures resulting in different overall QA performances. With the linguistic resource-side attributes - expressiveness and coverage - being the same for all of the QA systems, the method of deploying the resources is the key in determining the performance of QA systems. These QA studies, however, indicate that there is no agreement on which resource can provide the best external knowledge for answer processing. On the other hand, having accepted that each resource provides different types of linguistic information, many QA systems use a combination of them to take full advantage of their linguistic knowledge at different levels and steps. As a result, the key questions are: i) how to deploy the different resources in order to obtain full advantage of their linguistic information, and ii) how to implement a method to combine the results acquired from each resource.

It seems that FrameNet is one of the resources that has not been studied sufficiently in the context of QA to date and is attracting more attention recently. The type of semantics that it provides along with the unique frame-based generalization - semantic normalization - over the different part-of-speech predicates makes it a suitable resource for different parts of a QA system. Although the different factors mentioned earlier play important roles in linguistic QA and more specifically in FrameNet-based QA; however, they have not been carefully studied yet. Such studies, with respect to any natural language application, especially QA, could have developed an informative platform to:

- Develop the linguistic resources, and/or their counter-part systems that contribute linguistic knowledge to texts (with emphasis on applications),

- Distinguish the effectiveness and efficiency of the different linguistic resources with specific attention to certain applications, and
- Identify the bounds (especially the upper bound) of the contributions that each linguistic resource, with its current properties and capabilities, can provide to the different parts of natural language applications.

There are very few studies to provide such insight on linguistic resources. As one of the rare works in this direction, the informative role of WordNet in the context of QA is studied in (Paşca and Harabagiu 2001). In the sense of FrameNet, the only related work, as discussed in section 2.2.3, can be found in (Shen and Lapata 2007) where the importance of FrameNet semantic roles in factoid QA is studied. However, it does not cover other related aspects discussed in this section. These aspects directly affect linguistic answer processing modules, especially those relying on the FrameNet-based approaches.

2.3 Research Problems

The research problems in this thesis are concerned with two main parts of factoid linguistic QA systems discussed in the previous sections:

- Passage retrieval - the two main research questions in this part include:
  - How to use linguistic knowledge and non-linguistic features (such as density-based information of query terms) to enhance a passage scoring and ranking algorithm and elevate the effectiveness of answer passage retrieval in QA?
  - How to linguistically boost the passage retrieval process by using scenario-based relations in FrameNet at the input stage of passage retrieval to formulate the best query which maximizes the answer passage retrieval effectiveness?
- Answer processing - this consists of answering the following research questions:
  - How do the different levels of shallow semantic parsing with frame semantics encapsulated in FrameNet affect answer identification performance? What are the different contributions of the individual tasks of frame evocation and FE assignment to the task of factoid answer processing at different levels of annotation? And what is the role of different part-of-speech predicates in enhancing factoid answer processing performance?
  - How is the effectiveness of a frame semantic-based answer processing module affected by different techniques of semantic alignment using FrameNet entities? What is an optimal deployment method of FrameNet in factoid answer processing using semantic alignment?
  - What is the effect of the FrameNet coverage over different predicates on the frame semantic-based answer processing task? Can this be quantified in terms of the factoid answer processing performance?

- How is the overall answer processing accuracy influenced by different techniques of answer list merging in the presence of a frame semantic-based answer processing module and another (entity-based) module? Why is it important to fuse frame semantic-based and other models of answer processing? What is the upper bound of the answer processing effectiveness when fusing these models?

2.3.1 Enhancing Answer Passage Retrieval for QA Using Linguistic Information

In order to fill in the research gaps identified in section 2.1.4, on improving answer passage retrieval for QA, we aim to enhance passage retrieval methods so that they are capable of retrieving more answer passages and can deal with different types of deep scenario-based relations between the question and passage terms. To improve the passage retrieval methods, we focus on two aspects of passage retrieval methods: i) scoring and ranking algorithms, and ii) input analysis.

A ranking algorithm scores the text snippets that are identified by the retrieval and matching algorithms. Subsequently, it ranks the passages and reports the top-ranked ones. Input analysis aims to guide the retrieval algorithm with the best input query to return the most specific passages in response to a natural language question.

In the case of the ranking algorithm, the major research question is how to enrich an existing well-established passage retrieval method in order to obtain more specifically related passages. In the context of QA, we would prefer to have the smallest possible number of returned passages for answer processing to reduce the burden of the final answer extraction and scoring task and increase its accuracy. Therefore, it is crucial for the passage retrieval module to retrieve correct answer passages with high ranks.

In terms of input formulation, the main question looked at in this study is how to boost the retrieval process by exploiting FrameNet, which encapsulates scenario-based lexical relations, at the initial steps of passage retrieval. A specific type of query rewriting based on scenario-based relations in frame semantics will be examined to decide wether it can enhance passage retrieval performance.

2.3.2 Frame Semantic-Based Factoid Answer Processing

The discussion on the key aspects of using different linguistic resources for factoid answer processing leads us to analyse the impact of different levels of shallow semantic parsing, with FrameNet elements, on factoid answer processing performance. We will investigate what level of parsing may be required to reach high levels of answer processing performance. We will also see what part-of-speech predicates may play important roles in more effectively answering factoid questions using frame semantic alignment. This will be helpful when considering future analyses of developing FrameNet through improved FrameNet-based QA.

In addition, we will analyse the sensitivity of answer processing performance to the two subtasks of shallow semantic parsing, namely semantic class (FrameNet frame) identification and semantic role (FrameNet FE) labeling. The outcome of this part of the study will be useful for improving automated shallow semantic parsers to more effectively take part in QA.

We will also develop a number of different techniques for frame semantic-based alignment of factoid questions and answer passages to identify answer candidates and score them. In doing this, we will study the performance of a range of techniques of answer processing using frame and/or FE matching between question and passage frames and FEs. We will conceptually analyse these techniques and especially investigate the level of semantics that is taken by each technique. By analysing and proving which technique performs best in our experiments, we will be able to conclude what level of semantics is optimal (subject to highest answer processing performance) to be considered by these techniques so far.

The FrameNet lexical coverage over different part-of-speech predicates is another subject of analysis in our work. This will shed more light on the ways of improving FrameNet so that significant improvements in factoid FrameNet-based answer processing performance will be possible. We consider the results of our work on the importance of different part-of-speech predicates in answer processing performance and put this together with the outcomes of the analysis of FrameNet coverage and its impact on the accuracy of answer identification and scoring. Consequently, we will draw conclusions on which predicates are in a crucial stage of development in FrameNet with respect to the task of QA.

With some existing limitations in sole usage of FrameNet for answer processing, it is possible to make use of hybrid answer processing models that rely on different linguistic resources and/or approaches of answer extraction and scoring. We will investigate the process of merging answer lists obtained by a frame semantic-based answer processing model with those extracted by an entity-based model. We will propose two methods of answer list fusion that merge results according to their scores and ranks. In this part of our work, we will also show why it is necessary to fuse frame semantic-based and other models of answer processing to obtain improved performances.

2.4 Summary

Among many existing QA systems with different approaches, those which utilize linguistic information have been shown to achieve greater performances. Therefore, the different methods and approaches of using linguistic information in QA systems have been explored in this chapter. More specifically, the two sub-processes of passage retrieval and answer processing have been under consideration with a focus on the extent to which the linguistic knowledge at various levels is exploited.

In the passage retrieval part, issues like the level of linguistic knowledge that can be used for more effectively retrieving answer passages, the passage boundary detection techniques, the passage

indexing-based methods, and online analyses have been discussed which cover a broad range of retrieval methods and systems.

In the answer processing section, however, the review of existing works was based on the different linguistic resources and structures that have been used to contribute linguistic knowledge to the process of answer candidate detection and scoring.

For each of the areas of passage retrieval and answer processing, key research problems have been identified. In the case of passage retrieval, a way of enhancing answer passage retrieval by analysing the passage scoring and ranking function in a baseline algorithm is to be studied. In addition, a linguistic approach to boost the effectiveness of answer passage retrieval by the input query analysis has been identified to be another concern of this study.

For answer processing, the analysis of a linguistic resource - FrameNet - is the main concern of our research since there has been limited work on this previously. We will investigate the impact of shallow frame semantic parsing, frame semantic alignment technique, FrameNet lexical coverage, and answer processing models fusion technique on the answer processing performance of a factoid QA system.

Chapter 3

Methodology

The overall methodology for addressing the research problems (identified in Chapter 2) on linguistic passage retrieval and answer processing in factoid QA systems is described in this chapter. Testing our hypotheses on linguistic FrameNet-based passage retrieval and answer processing requires well-defined settings and a managed platform to effectively quantify the improvements that can be achieved by employing the frame semantics elements encapsulated in FrameNet. Since passage retrieval and answer processing tasks have different characteristics and requirements, their methodological aspects are explained in two separate sections. For both parts, this includes an overview of how the study is conducted and a description of the experimental settings. The data used in the experiments, baseline systems, and evaluation criteria are also explained for each part.

3.1 Answer Passage Retrieval in QA

For answer passage retrieval in QA, there are two main research questions being studied in this thesis:

i) How to enrich a passage retrieval method with a passage scoring and ranking algorithm using linguistic and non-linguistic features to obtain more relevant answer passages in top ranked passages? and
ii) How to linguistically boost a passage retrieval method by formulating the best retrieval query using the frame semantics in FrameNet?

An overview of the methodology of answering these two questions is given in the next section followed by an introduction to the data, baseline passage retrieval methods, and evaluation metrics.

3.1.1 Enhanced Passage Retrieval Methods

To improve a passage retrieval method with an enhanced passage scoring and ranking algorithm, we use linguistic information at the syntactic level, term density information, passage lengths, and query term coverage (the number of terms occurred in a given passage). We take the MultiText passage retrieval algorithm (Clarke, Cormack, and Burkowski 1995; Clarke et al. 2000; Clarke, Cormack, and Tudhope 2000; Cormack et al. 1998) and enhance its retrieval effectiveness in the context of the TREC QA task. A passage scoring (question-passage similarity) function is proposed and plugged into the MultiText algorithm to more effectively score and rank retrieved passages. The function is dependent on our question and passage representation procedure.

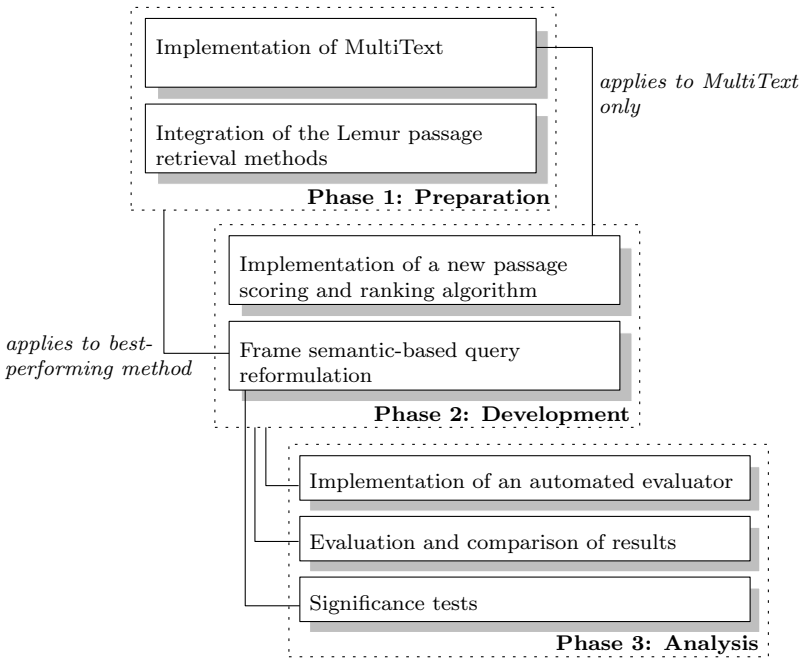

Figure 3.1: Schematic view of study for linguistic passage retrieval in QA

In answering the second research question, the input query analysis, we semantically boost the retrieval effectiveness of a passage retrieval algorithm. The boosting procedure, in an iterative way, accesses the frame semantics knowledge on the English predicates encapsulated in FrameNet to overcome the deep surface mismatches between similar concepts in questions and answer-bearing passages. The iterative query rewriting process converges to the best query that maximizes the number of answer passages in the list of the top n passages retrieved per question. This is addressed by analysing the scores of the passages retrieved per query in each iteration. The baseline non-linguistically boosted passage retrieval method, on each dataset, is selected. This is the best-performing retrieval method among the set of methods under consideration in our experiments. The

set includes the Lemur passage retrieval methods which will be described in section 3.1.3 and the MultiText algorithm with both the original and contributed passage scoring and ranking strategies. The reason for selecting the best-performing passage retrieval method for boosting is to test if frame semantic-based input analysis can improve the answer passage retrieval performance of the best baseline of retrieval achieved with no FrameNet-based query analysis.

Figure 3.1, sets out the methodology for studying the two research questions in passage retrieval. It consists of three main phases: i) preparation, ii) development, and iii) analysis. In the first phase, the required software implementations are carried out and in the second phase new ideas to address the research questions are developed and added to the baseline systems. In the last phase - analysis - the results and the observations are analysed to draw conclusions. Chapter 4 details our study on enhancing answer passage retrieval in QA.

3.1.2 Data

The datasets under experiment are the TREC 2004 and TREC 2006 factoid question sets and their corresponding text collection - AQUAINT[1]. This collection contains the news articles from the New York Times News Service (1998-2000), Xinhua News Service (1996-2000), and Associated Press Worldstream News Service (1998-2000).

The TREC 2004 question set contains 65 targets and 230 factoid questions. We run the passage retrieval methods on a subset of 208 questions for which there exists an answer in the AQUAINT document collection. There are 22 questions in the 2004 track with no answers in the collection according to the TREC report (Voorhees 2004). The set of 208 TREC 2004 questions is used to tune and train the retrieval methods with the TREC specifications and requirements. It is also used for training and setting up the algorithms which are developed to address the passage retrieval-oriented research questions in this thesis. The NIL-answer questions are removed from the experiments because we need to have answers for passage retrieval evaluations.

In the TREC 2006 track, there are 403 factoid questions grouped under 75 different targets (Dang, Lin, and Kelly 2006). We run the experiments on 386 factoid questions in this set as there are 17 questions with NIL answers. The questions in this set are used only for testing the algorithms under study.

The passage retrieval methods are not run on the whole set of the AQUAINT collection. In contrast, the set of related documents retrieved by the PRISE search engine and reported by TREC for each target, including 50 documents per target, is defined to be the document set for each question[2]. This reduces the burden of implementation and running the document retrieval task on the massive set of documents in the AQUAINT collection with 1,033,461 documents which would require a huge index as well, especially in the case of the MultiText passage retrieval algorithm

[1] http://www.ldc.upenn.edu/Catalog/docs/LDC2002T31/
[2] Queries belonging to a TREC target share one index.

which needs to have access to the term positions in the documents. Since this reduction is carried out for all of the passage retrieval methods in our experiments, there will be no methodological bias towards answers for any of the retrieval methods. As a result, the evaluations will not be negatively affected.

The input questions, in the case of the second research problem in passage retrieval - the frame semantic-based analysis of input queries - are semantically annotated in accordance with the procedure that will be explained in Chapter 4.

3.1.3 Baseline Passage Retrieval Systems

The MultiText passage retrieval algorithm (Clarke, Cormack, and Burkowski 1995; Clarke et al. 2000; Clarke, Cormack, and Tudhope 2000; Cormack et al. 1998) and the Lemur passage retrieval methods[3] are used in the passage retrieval experiments conducted.

The Lemur passage retrieval package includes a series of retrieval methods which will be explained later. This makes the Lemur package suitable for the purpose of evaluating a specific passage retrieval method with a list of different well-known passage retrieval methods that interpret a passage as a fixed-length sequence of words.

On the other hand, MultiText is one of the best-known passage retrieval algorithms which have been exploited for document ranking and retrieval purposes as well as the passage retrieval task. This algorithm interprets all textual documents as a continuous series of words and also interprets passages as any number of words starting and ending at any position in the documents. A document d is treated as a sequence of terms $t_1, t_2, \ldots, t_{|d|}$ and the query is translated into an unordered set of terms $Q = \{q_1, q_2, \ldots, q_{|Q|}\}$. There are two concepts that need to be defined:

- An *extent* over a document d is a sequence of words in d which contains a subset of query terms. It is denoted by the pair (p, q) where $1 \leq p \leq q \leq |d|$ given that p and q are term positions/offsets in document d. This translates into the interval of texts in document d from t_p to t_q. An extent (p, q) satisfies a term set $T \subseteq Q$ if the extent includes all of the terms in T.
- An extent (p, q) is a *cover* for the term set T if it satisfies T and there is no shorter extent $(\acute{p}, \acute{q})$ over the document d which satisfies T. A shorter extent $(\acute{p}, \acute{q})$ is a nested extent in (p, q) where $p < \acute{p} \leq \acute{q} \leq q$ or $p \leq \acute{p} \leq \acute{q} < q$. In any document d there may be different covers for T which are represented in the cover set C for the term set T.

The passages retrieved by MultiText are identified by covers; therefore, they start and end with pairs of the query keywords and have variable lengths. Covers do not overlap the document boundaries in the unique string of words and sentences of the whole document set which is a requirement for retrieving actual passages in the documents. The passages retrieved by this algorithm are scored based on the length of the passages and the weight of the query terms covered in the passages. Each

[3] http://www.lemurproject.org/

term t gets the IDF-like weight as shown in Equation 3.1, where f_t is the frequency of the term t in the corpus and N is the total length of the unique string constructed over the document set.

$$w_t = log(\frac{N}{f_t}) \tag{3.1}$$

A passage containing a set T of the terms is assigned a score according to the formula in Equation 3.2 where p and q are the start and end points of the passage in the unique string of words in the document set.

$$Score(T, p, q) = \sum_{t \in T} w_t - |T| log(q - p + 1) \tag{3.2}$$

Experiments performed by Tellex et al. (2003) show that MultiText performs well; the third highest mrr in the documents retrieved by the PRISE search engine and the highest mrr in those retrieved by the Lucene[4] search engine. The results provide a comparison among the eight passage retrieval algorithms investigated including MITRE (Light et al. 2001), bm25 ((Robertson et al. 1995), MultiText, IBM (Ittycheriah, Franz, and Roukos 2001), SiteQ (Lee et al. 2001), Alicante (Llopis and Vicedo 2001), ISI (Hovy, Hermjakob, and Lin 2001), and Voting (Tellex et al. 2003). The high performance of MultiText, as well as its frequent participation in TREC (Clarke et al. 2000), is the main reason for choosing MultiText as one of the passage retrieval algorithms in our experiments. The Lemur toolkit contains the other passage retrieval methods used in the experiments. Lemur is a toolkit designed to facilitate research in language modelling and information retrieval. It includes a set of well-designed and supported Application Programming Interfaces (APIs) for text indexing, retrieval, summarization, and clustering. We use the Lemur toolkit for two purposes:

i) Indexing top ranked documents per TREC target for the MultiText passage retrieval algorithm keeping the term positions. Table A.1 (Appendix A) shows the parameter set.
ii) Indexing fixed-length passages and retrieving these passages with the different Lemur retrieval models. Table A.2 (Appendix A) summarizes the parameter set for indexing passages.

Focusing on passage retrieval, Lemur has a set of retrieval models each of which can be applied for both ad hoc document retrieval and passage retrieval tasks. The models that we use in our passage retrieval experiments include:

- TF/IDF
- Okapi BM25
- CORI collection selection
- Cosine similarity
- InQuery-CORI
- KL-DivergenceLanguage

[4] http://lucene.apache.org/

The task of passage retrieval in Lemur is performed based on fixed-length passages in the documents, while passages have overlaps equal to half of the fixed length of the passages. We set the size of the passages to 300 words to be consistent with the optimum range of passage lengths from 150 to 300 words mentioned in (Kaszkiel, Zobel, and Sacks-Davis 1999). The Lemur toolkit version 3.1.2 is used in our experiments. The parameter set for retrieving passages using the Lemur retrieval models is provided in Table A.3 (Appendix A).

3.1.4 Evaluation Metrics

As explained in (Kaszkiel, Zobel, and Sacks-Davis 1999) the two aspects for evaluating a text retrieval system - passage retrieval in this case - are *efficiency* and *effectiveness*. The former measures the usage of the resources such as disk, time, and memory, while the latter is concerned with the satisfaction level of users by retrieved texts. In the context of QA, the effectiveness of the retrieval is considered to be more important especially to the extent that the retrieval units need to potentially contain the correct answers to the natural language questions.

Focusing on the TREC QA track, our judgment of the passages, after the tasks of retrieval and ranking are accomplished, is based on whether the retrieved passages satisfy the correct answer patterns reported by TREC for each question. In standard passage retrieval, passages are judged for *relatedness* or *aboutness*; however, in this paradigm of retrieval we are more rigorously assessing passages on whether or not they contain correct answers. This has been referred to as *specificity* in Chapter 2. Specificity is a more stringent requirement than relatedness; consequently, many highly related passages which do not have actual answers will fail from a specificity point of view.

Manual evaluation of passage retrieval methods, with respect to specificity of passages, is an intensive task because of:

- **Multiplicity of answer patterns:** The evaluator needs to search for each answer pattern of a given question in a set of top-ranked retrieved passages and record the rank of the first answer-containing passage for each answer pattern,
- **Multiplicity of retrieval methods:** There are a number of passage retrieval methods to be evaluated, and
- **The size of question sets:** There are a large number of questions to be evaluated in two different question sets (TREC 2004 and TREC 2006 factoid questions).

We implement an automated passage evaluator to ease the process of searching for answer patterns in top-ranked passages. We use this software across all of the passage retrieval methods under experiment.

Our software evaluator looks for string occurrences in passages. Therefore, to enable this evaluator to identify answer occurrences, we first manually convert TREC-reported answer patterns (in the form of regular expressions) to plain texts. For example, the answer pattern "(auto|car) crash" is

split to two answer strings "auto crash" and "car crash". Then the software evaluator matches the passages with both of the answer strings and decides whether a given passage contains any of the answer strings or not. The main reason for splitting the TREC-reported regular expressions into plain strings is to enable the software evaluator to recognize which answer string is satisfied by a given passage. This is a crucial requirement in evaluating the retrieval methods on evaluation metrics explained later.

This software can identify most answer occurrences in the passages. A partial evaluation of this software shows that for the modified MultiText-retrieved passages, the software can identify an answer for 126 questions out of 230 factoid questions in the TREC 2004 question set in the list of the top 10 passages with *strict* evaluation (see the next paragraph). However, the manual evaluation of the same set of retrieved passages results in 139 questions with an answer found in the list of the top 10 passages. This shows an accuracy of ∼90% for our software passage evaluator. A similar evaluation shows an accuracy of ∼93% for the original MultiText-retrieved passages for the TREC 2004 questions. This provides evidence for our software evaluator not being biased towards any of the retrieval methods.

Both *strict* and *lenient* evaluation paradigms are considered in the experiments. In strict evaluation, it is necessary for the correct answers to have been extracted from a short list of related documents to a question as reported by TREC. In the lenient evaluation method, however, answers can be retrieved from any document in a larger set of documents related to the question (this set contains 50 documents per TREC target/question).

The main metrics for evaluating the effectiveness of the passage retrieval tasks in our experiments include:

- **Accuracy:** This is the rate of finding correct answers per question at the top *Rank* passages ($acc@Rank$ where $Rank \in \{10, 15, 20\}$). Accuracy gives an overall understanding of the maximum number of questions that could possibly be answered if a particular passage retrieval method was used. Equation 3.3 shows the formula for measuring accuracy where n_q is the total number of questions, and $af_i@Rank$ indicates whether at least one of the answers for the question q_i is found in the top *Rank* passages.

$$\begin{aligned} acc@Rank &= \frac{1}{n_q}\sum_{i=1}^{n_q} af_i@Rank \\ af_i@Rank &= 0; \textit{no answer found in the top Rank passages} \\ af_i@Rank &= 1; \textit{an answer found in the top Rank passages} \end{aligned} \tag{3.3}$$

- **Mean Reciprocal Rank (*mrr*):** It is calculated using Equation 3.4 where n_q is the total number of questions and $ar_i@Rank$ stands for the rank of the first answer-bearing passage for the question q_i in the top *Rank* passages ($mrr@Rank$ where $Rank \in \{10, 15, 20\}$). From a

QA point of view, mrr plays an important role since the answer processing procedure can be highly dependent on the rank of the answer-containing text snippets.

$$mrr@Rank = \frac{1}{n_q} \sum_{i=1}^{n_q} \frac{1}{ar_i@Rank} \tag{3.4}$$

- **Average precision - average recall:** These are standard measures used in information retrieval. The average values of precision and recall are calculated over the set of questions. We do not calculate precision values at standard recall levels; instead, the precision values are evaluated at the level of the top *Rank* passages retrieved ($prec@Rank$ where $Rank \in \{10, 15, 20\}$). The main reason for this is the importance of measuring the appearance of answer-containing passages at high ranks. Therefore, our focus is on a limited number of the top-ranked passages instead of the distribution of precision at a range of standard recall levels. We also measure recall values at the level of the top *Rank* passages ($rec@Rank$ where $Rank \in \{10, 15, 20\}$).

$$avg\_prec@Rank = \frac{1}{n_q} \sum_{i=1}^{n_q} \frac{n^i_{abp}}{Rank} \tag{3.5}$$

$$avg\_rec@Rank = \frac{1}{n_q} \sum_{i=1}^{n_q} \frac{n^i_{am}@Rank}{n^i_a} \tag{3.6}$$

The method to calculate the recall and precision measures for each question considers the answer set which contains the regular expression answer patterns reported by TREC. Calculation of precision values is based on passages; however, measuring recall values is based on answers. This is because the set of correct answer-bearing passages per question is not known and therefore, recall values cannot be measured based on passages. Equation 3.5 and Equation 3.6 show the formulas for measuring average precision $avg\_prec@Rank$ and average recall $avg\_rec@Rank$ respectively in the top *Rank* passages where $n^i_{am}@Rank$ is the number of unique answers occurring in the set of the top *Rank* retrieved passages for the question q_i, n^i_a is the total number of correct answers for this question, n^i_{abp} refers to the number of answer-bearing passages, and n_q indicates the total number of questions.

Selection of the three levels of retrieved passages ($Rank \in \{10, 15, 20\}$) is used to appreciate the performance of algorithms at different levels of passage retrieval. These three levels are appropriate for the needs of a natural language QA system that performs intensive text understanding processes in the answer processing phase. At the same time, they contain enough related sentences ($\sim$25, $\sim$38, and $\sim$50 sentences in the top 10, 15, and 20 passages respectively) from which the candidate and actual answers can be extracted. These passages are also shown to contain a reasonable number of correct answers according to our analysis of the TREC 2004 and 2006 factoid questions. Figure 3.2 shows the number of questions with at least a single correct answer (with strict evaluation) retrieved

on each passage retrieved by our modified MultiText and ranked from 1 to 20. The number of correct answers has a decreasing trend with the rank of passages in both the TREC 2004 and 2006 datasets. Therefore, there is only a small chance of retrieving correct answers in passages which would be ranked lower than the 20th passage.

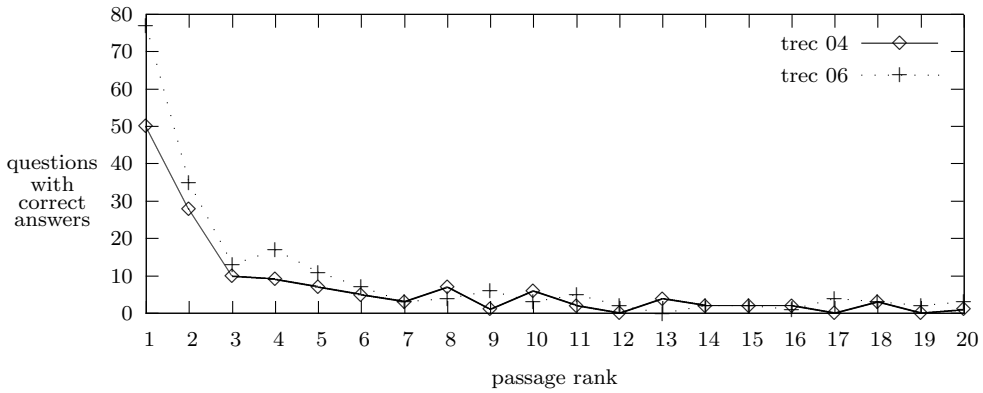

Figure 3.2: The number of questions with at least a single correct answer on each passage rank (1 to 20) retrieved by our modified MultiText for factoid questions in the TREC 2004 and 2006 datasets

The paired t-test is conducted on the passage retrieval evaluations to assess the statistical significance of the results obtained for our passage retrieval-based tests. For enriching passage scoring and ranking functions (the first passage retrieval-based research problem), this includes the significance test between the results of our modified MultiText and other methods. In the case of linguistically boosting passage retrieval with FrameNet (the second passage retrieval-based research question) the test is performed between the linguistically boosted method and its non-linguistically boosted version.

To calculate the paired t-values, we use the performance measure of the two methods (regarding each evaluation metric) at the level of top *Rank* passages for each single question. These individual (question-based) performance measures are then used for calculating paired t-values. The paired t-values are finally mapped to statistical p-values where any value less than 0.05 will indicate a statistically significant result.

3.2 FrameNet-Based Factoid Answer Processing

In this section, we discuss the methodology employed in addressing research questions on answer processing in factoid QA. In the following chapters, we will show that the usage of frame semantics in factoid QA can develop the effectiveness of factoid answer processing beyond that of named entity-based approaches that interpret answers as succinct named entities in texts. To demonstrate this,

we test the impact of the following aspects on factoid answer processing performance:

- The shallow semantic parsing level,
- Frame semantic alignment technique,
- FrameNet lexical coverage, and
- The method of fusing answer lists of two answer processing models.

There is a challenge in using FrameNet (and other similar linguistic resources) for natural language applications which is known as Word Sense Disambiguation. This is concerned with finding the correct semantic class that defines a target predicate. For example, the word "make" has several senses in English such as "constructing", "cooking", and "arriving" and it is necessary for systems to identify the right sense of a certain occurrence of this predicate in a certain sentence or paragraph. In this thesis, we do not put emphasis on this as we use an automated shallow semantic parser (see section 3.2.4) and then manual corrections are carried out on the results of automated parsing.

In the next section, an overview of the methodology employed in testing different aspects of FrameNet-based answer processing is given. This is followed by a description of question sets under experiment, the experimental QA system, a baseline shallow semantic parser for frame semantic-based text annotation, a manual annotation tool, baseline QA systems, and the evaluation metric for analysing the effectiveness of different answer processing runs.

3.2.1 Experimental Setup for Evaluating FrameNet-Based Answer Processing

In answering these research questions, we use our implemented experimental QA system which will be explained in detail in section 3.2.3. This QA system is particularly used for practical justification of our research outcomes and comparing our methods with the baseline systems described in section 3.2.6.

In conjunction with the entity-based answer processing model in our experimental QA system, a frame semantic-based model is implemented that identifies answer candidates by performing frame semantic alignment on frame semantically annotated questions and passages. Figure 3.3 shows the general methodological steps towards answering the above-mentioned research problems. This involves four major phases: i) preparation, ii) development, iii) tuning and pre-testing, and iv) analysis.

The phase of preparation includes the initial activities necessary for starting the study while the second phase - development - consists of the theoretical analysis and proposal of new techniques, tools, and approaches. Development starts with the requirement analysis of the research problems with respect to the frame semantic-based answer processing followed by further theoretical improvement in each part. The other phases of tuning, pre-testing, and analysis mainly carry out the practical QA runs, evaluations, interpretations of the results, and drawing conclusions. These will

be explained in detail from Chapter 5 to Chapter 8.

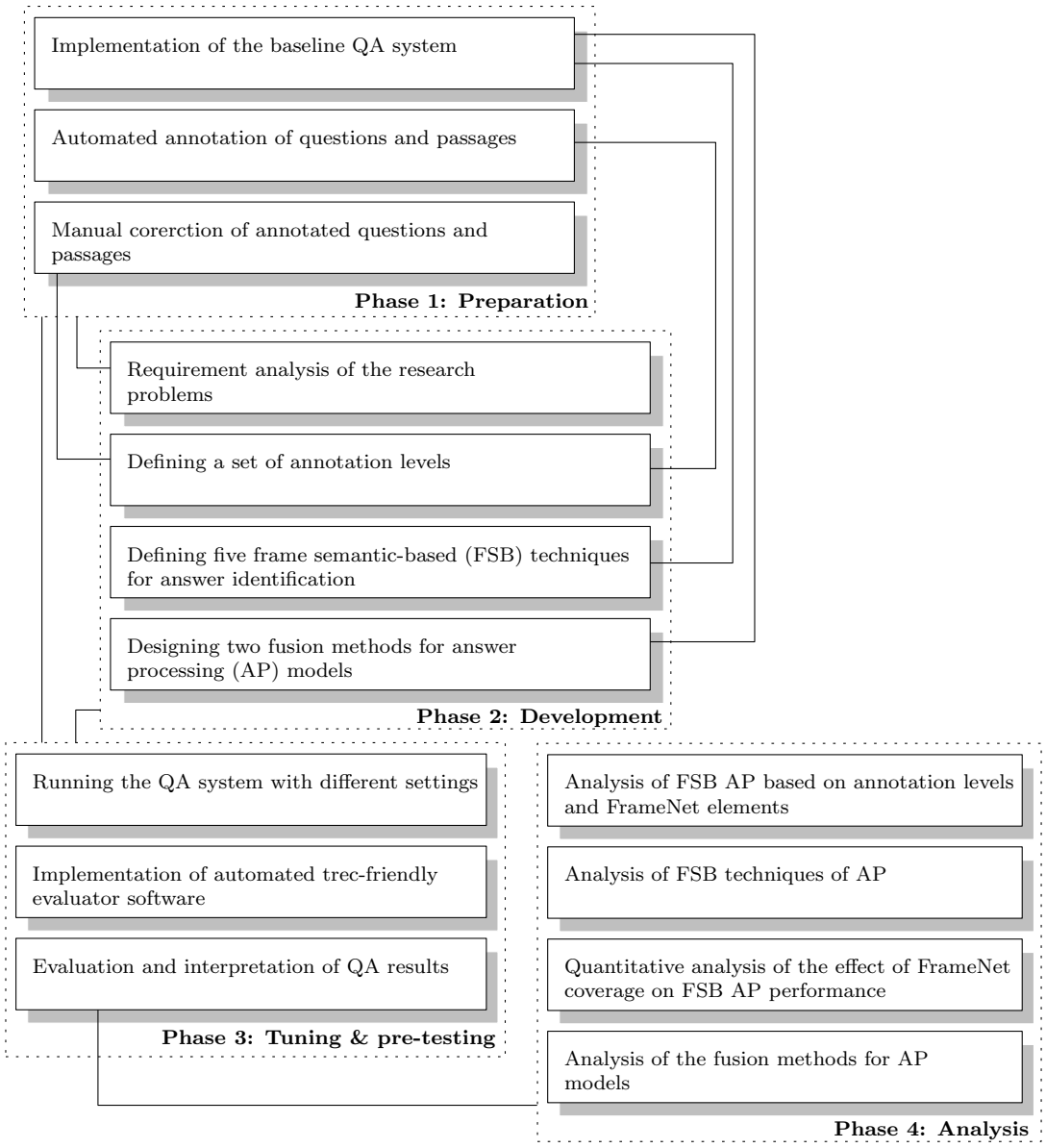

Figure 3.3: The hierarchy of the general activities to study the research problems in linguistic answer processing

The schematic view of the different experiments that we conduct in tackling the answer processing problems is illustrated in Figure 3.4. The Base experiment includes running the experimental QA system with the entity-based answer processing model. The results of the Base experiment will be

used in Experiment 1 and Experiment 4.

To analyse the impact of different levels of annotation on the system, in Experiment 1, the fully automated annotation outputs of a baseline shallow semantic parser (see section 3.2.4) are manually corrected and four levels of annotation are considered (see Chapter 5). The different contributions of correct frame labeling and FE assignment are also recorded at these levels of annotation as well as the overall effect of more sophisticated annotation on the performance of factoid answer processing.

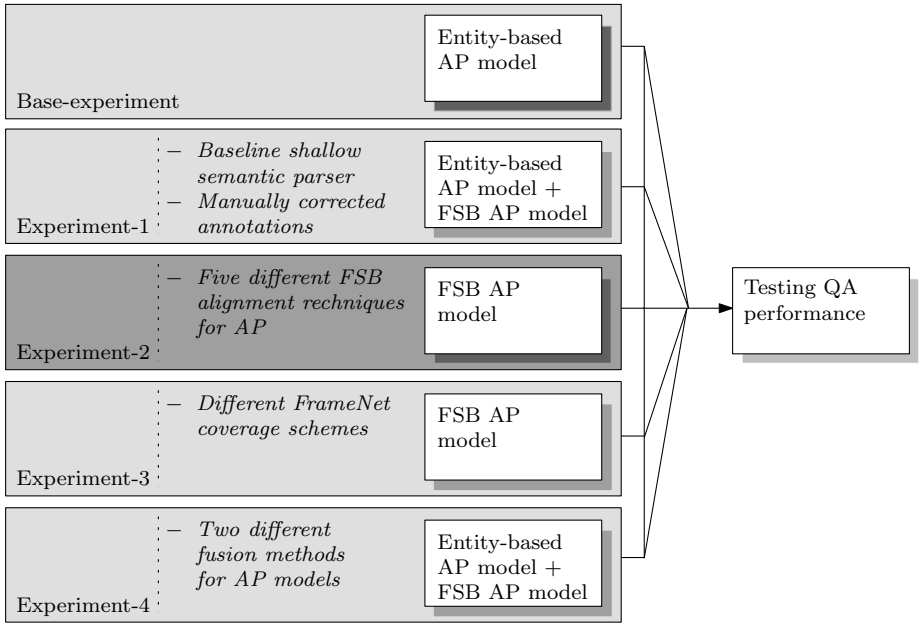

Figure 3.4: Different experiments to address the factoid answer processing research problems - Experiment 2 is the main experiment of the thesis

In Experiment 2, which is the main experiment in this thesis where we propose new and effective FrameNet-based answer processing methods, different techniques of frame semantic alignment between questions and passages are implemented in the frame semantic-based answer processing model. The effectiveness of these different answer processing methods (using FrameNet elements) is measured by testing the performance of the QA system in different runs. The techniques are all based on frame and/or FE alignment to identify answer candidates. Chapter 6 details the baseline frame semantic-based answer processing method and the new methods that we have developed. The techniques used in these methods range from a complete frame and FE alignment strategy to a shallow FE-based alignment that ignores the big semantic pictures of FrameNet frames. By running the frame semantic-based answer processing model with these different techniques, results for each answer processing strategy are generated and evaluated to identify the best-performing strategy.

To test the positive impact of higher levels FrameNet coverage over different English predicates on the performance of the frame semantic-based answer processing, in Experiment 3, different versions of the FrameNet dataset - the FrameNet 1.2 and FrameNet 1.3 datasets - are used for annotating texts and identifying answer candidates. The different versions in the FrameNet dataset have an increasing coverage over time, which may affect different natural language applications. We will test how FrameNet coverage affects factoid answer processing performance. By analysing experiments that we conduct, it will be inferred which type of predicates play a more important role in QA and require more coverage-related work in FrameNet. Chapter 7 explains related issues.

Experiment 4 analyses the impact of the fusion method of a frame semantic-based answer processing model with a non-semantic entity-based model on the performance of the frame semantic-based model and the overall performance of the QA system. Two methods of answer list merging are considered in our experiments: i) the score-based fusion method, and ii) the rank-based fusion method both will be explained in Chapter 8. The QA system, with the two answer processing models - the frame semantic-based and entity-based models, is run on the different question sets and the results of the merging methods are compared in the sense of the number of correct answers retrieved. After performing these experiments on fusion techniques, more tests are carried out at a lower level which focuses on the fusion parameter. The parameter-level test includes analysis of the score-based fusion technique with respect to its internal convex parameter that is used to set the emphasis on the answers of each answer processing model. This test examines the possibility of retrieving a greater number of correct answers by each model. This is important to the extent that a challenge in the fusion task is how best to combine the answer processing models so that the maximum number of questions are correctly answered.

3.2.2 Data

The two factoid question sets described in section 3.1.2 - the TREC 2004 and TREC 2006 factoid question sets - and the AQUAINT document collection are used as the question sets and the answer resource for analysing the different QA runs. The question sets, however, are filtered down to the subsets that satisfy pre-defined and post-defined conditions necessary for our answer processing experiments.

Table 3.1: The filtering scheme of the experimental question sets

Dataset	Total # of items	No answer @10 passages	N/A after frame semantic-based analysis	Remaining
trec04 factoid question set	230	87	68	75
trec06 factoid question set	403	227	N/A	176

Table 3.1 shows the filtering figures of the two question sets. In the TREC 2004 question set, questions for which no answer can be extracted from the top 10 passages are removed from the experimental question set (Column 3). There are more limitations imposed with respect to a frame semantic-based analysis on the questions and answer passages which will be explained in detail in Chapter 5 (Column 4). The limitation of the passage lists to contain a maximum of 10 passages per question is due to the intensive automated and manual annotation task in the TREC 2004 question set. For the TREC 2006 set, filtering is just based on the evaluation of the passage retrieval task to have correct answers in the top 10 passages.

In both question sets (TREC 2004 and TREC 2006), removing questions for which no answer can be extracted in the top 10 passages is based on a strict evaluation of the passages retrieved by our modified MultiText algorithm (see Chapter 4). In the TREC 2004 task, our modified MultiText algorithm is semantically boosted which is not the case in the TREC 2006 task (see section 4.2.2 in Chapter 4). The other difference is that in the 2004 question set, the evaluation of the passages is performed manually while in the case of the 2006 dataset this is carried out using our implemented software evaluator described in section 3.1.4.

Table 3.2: The usage of question sets to study the research problems

Research problem in answer processing (AP)	Question set
The effect of different levels of FrameNet-based annotation on AP	trec04
The different effects of the frame labeling and FE assignment tasks on AP	trec04
The effect of semantic alignment technique on AP	trec04, trec06
The effect of FrameNet coverage on frame semantic-based AP	trec04, trec06
The effect of fusion method of a non-frame semantic-based model with a frame semantic-based model on the overall performance of AP	trec04, trec06

The usage of the datasets in different answer processing experiments is summarized in Table 3.2. The main reason for not using the TREC 2006 track for analysing the first two research questions is that it was not possible to perform manual annotation of the TREC 2006 track in the time frame of this thesis. This prevents the TREC 2006 dataset from being applicable for analysing the effect of the different levels of FrameNet-based annotation and the related subtasks of frame labeling and FE assignment on the performance of answer processing.

3.2.3 Experimental QA System

The pipelined architecture of our implemented QA system is shown in Figure 3.5. Once a question is analysed in the first module, the output information is passed on to the information retrieval module to retrieve the list of most related documents and answer passages. The passage-level information along with some pieces of question information are the inputs to the answer extraction and scoring module which finally reports the answers to a given question. The details of the three main modules

are given in the next subsections.

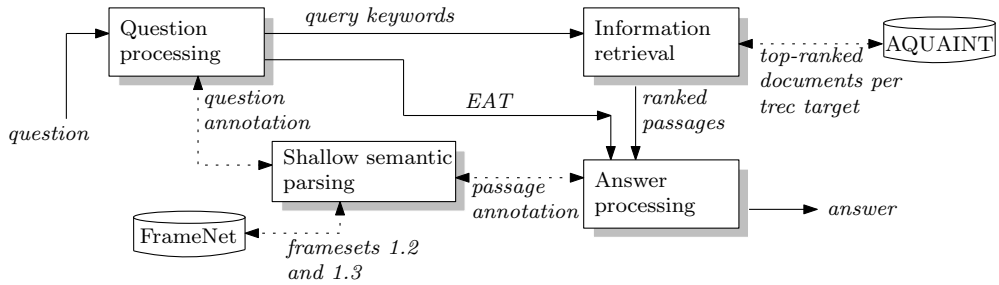

Figure 3.5: The pipelined architecture of our experimental QA system

3.2.3.1 Question Processing

In the question processing module, which is the first module that receives the question, there are three main tasks to be performed:

i) Hard classification of questions using a set of predefined classes,
ii) Identification of the Expected Answer Type (EAT) of questions, and
iii) Construction of information retrieval queries which will be exploited in the document and passage retrieval phase.

The question classification task in our QA system is performed using a shallow hand-crafted rule-base containing ∼130 rules to categorize the focus of the questions into one of the classes: PERSON, TITLE, LOCATION, TIME, ORGANIZATION, REASON, MANNER, PRICE, DATE, DEFINITION, NUMBER, MONEY, MONEY-NUMBER-DEFINITION-TITLE, TIME-DISTANCE, MONEY-PRICE, and UNKNOWN as described in (Moldovan et al. 2000). Our question classification rules rely on:

- The existence of specific question stems (such as *where*, *when*, and *why*),
- The n-grams with words as items (mostly bigrams, $n = 2$),
- The part-of-speech of terms, and
- The sequence of the occurrence of all of the items mentioned above.

The rule-base is constructed and trained on the TREC 2004 factoid question set and is tested with the TREC 2006 factoid question set. It achieves a classification accuracy of ∼98% on the training set and the accuracy of ∼76% on the test set. Since the training and test sets are totally disjoint (no intersections) and the TREC 2006 questions are more complicated than those in the TREC 2004 track, the accuracy of the rule-base is acceptable. This accuracy can have an influence on the effectiveness of the entity-based answer processing model (see section 3.2.3.3) where identification

Table 3.3: The mappings from question categories to NE types

Question category	Corresponding NE type selected as EAT
Money	Number
Number	Number
Definition	Definition
Title	Person, Organization
Person	Person
Organization	Organization
Date	DateTime
Location	Location
Manner	-
Time	DateTime
Distance	Number
Price	Number
Reason	-
Money-number-definition-title	Definition, Number, Person, Organization
Time-distance	DateTime, Number
Money-price	Number
Unknown	Number, Definition, DateTime, Location, Person, Organization

of EATs is crucial. If we carried out a manual classification of questions (with 100% accuracy), then the challenge of question analysis would be ignored. This is not, however, the case in real QA systems.

Identification of the EATs, however, is based on a lossy mapping from the fine-grained question classes to the coarse-grained set of EATs formed according to the set of Named Entity (NE) types that are supported in our QA system. We use the LingPipe NE tagger[5] to identify PERSON, LOCATION, and ORGANIZATION references in passages. In conjunction with these, we implement a pattern-based DATE-TIME tagger and NUMBER expression tagger. A simple DEFINITION tagger that identifies colours and a few general definitional adjectives is also implemented. Table 3.3 shows the mappings from the question classes to the NE types supported in our system which form the EATs of questions.

The other process in question analysis is to construct an information retrieval query for each question. This includes the following steps:

i) Stop-word removal using van Rijsbergen's stop-wordlist with very minor changes that we apply[6],
ii) Term stemming using the Porter stemmer to normalize terms to their roots, and
iii) Reference resolution in the questions which considers the TREC targets of the questions. If there is no explicit reference to the target concept of the questions, the target string is added to the query string. This is performed to ensure that the retrieved passages contain related information to the target topic of the questions.

[5] LingPipe: http://alias-i.com/lingpipe/

[6] The changes include removing the words "first", "found", "now", and "there" from the list as they add meaning to the questions in TREC and adding the word "did" to the list.

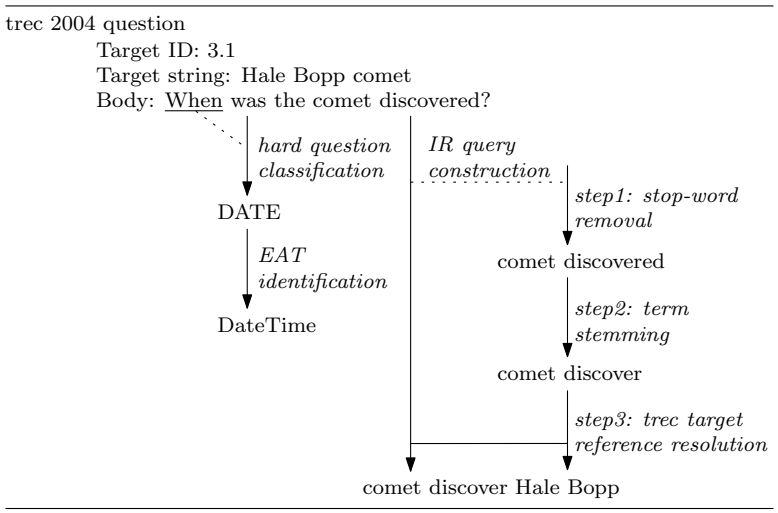

Figure 3.6: The question processing module takes three major steps

The three main steps of question analysis are shown on one of the TREC 2004 factoid questions in Figure 3.6.

3.2.3.2 Information Retrieval

The process of information retrieval is performed at the level of passages in the top ranked documents related to each TREC target reported by the PRISE search engine as part of the TREC-provided resources in the QA track. Therefore, our information retrieval module is limited to retrieving passages out of related documents without performing any document retrieval procedure. A modified version of the MultiText passage retrieval algorithm, which is part of the contribution of this thesis and will be discussed in Chapter 4, is used to retrieve passages for both question sets in the TREC 2004 and TREC 2006 datasets. Semantic annotation of these passages (to add semantic classes and their semantic roles to sentences) is an intensive task that requires much time and cost. To reduce the burden of this task, a maximum number of the top 10 retrieved passages per question are delivered to the answer processing module.

In the TREC 2004 dataset, our semantically boosted modified MultiText algorithm is used to retrieve passages. However, in the TREC 2006 task, our modified MultiText without any semantic boosting is exploited for retrieving passages.

3.2.3.3 Answer Processing

The answer processing module is implemented in a flexible fashion which employs two different models:

i) An entity-based model (ENB)
ii) A frame semantic-based model (FSB)

The flexible setting of the answer processing models allows for running the QA system with almost any possible order of the two models. Figure 3.7 shows this configuration.

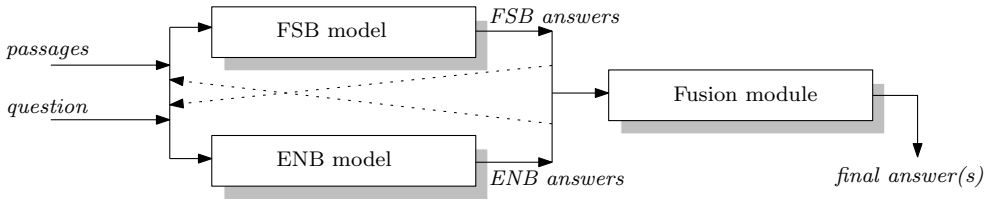

Figure 3.7: The flexible setting of the answer processing module

As shown in Figure 3.7, it is possible to obtain answers by the individual answer processing models and merge the result sets in order to report final answer(s). The different combinations of answer processing models include:

- **FSB-only:** to run the answer processing module with the FSB model only,
- **ENB-only:** to run the baseline answer processing with the ENB model only,
- **Combined (FSB-first):** to run the answer processing module with FSB and then ENB in case FSB fails to extract any answer,
- **Combined (ENB-first):** to run the ENB model first and the FSB model only if ENB fails to retrieve any answer,
- **Merged (FSB-ENB-fused):** to run both answer processing models and fuse their answer sets. The fusion (merging) strategies for the answer sets will be discussed in Chapter 8. In this setting, a correct answer may be attributed to either model. The overall performance of FSB-ENB-fused is equal to the summation of the individual performances of the FSB and ENB models.

Extraction of the answer candidates from the answer passages in the ENB model involves the following steps:

i) Extraction of the NEs from the retrieved answer passages,
ii) Filtering the set of NEs with respect to the EAT of the question (see Table 3.3), and
iii) Ranking the remaining NEs according to the score of the NE-bearing answer passages. Each NE receives the score of its passage (already calculated by the passage retrieval method) and finally all of the NEs are sorted with the highest-scored NE as the first answer.

As mentioned before, the set of NEs (and similarly EATs) include the PERSON, LOCATION, ORGANIZATION, DATE-TIME, NUMBER, and DEFINITION references. These references, however, cannot cover the answers to *why* and *how* questions. The *why* and *how* questions may be

answered by the FSB answer processing model as it is designed in such a way that it is not limited to a subset of NE types.

Our ENB model achieves an *mrr* value of 0.400 on a set of 75 factoid questions in the TREC 2004 dataset. There are eight TREC participants which achieve higher *mrr* values on the same subset of factoid questions. The best-performing system (LCC's QA system) achieves the *mrr* value of 0.867 on these questions. The performance of LCC's QA system is not due to a NE-based approach alone but both a high-accuracy NE tagger and a logic prover of lexical chains on the basis of WordNet relations (see section 2.2.1).

In the FSB model, however, there is no NE-oriented analysis; instead, both questions and retrieved passages are annotated with FrameNet frames and FEs using a shallow semantic parser. Having a vacant FE identified in the question, the process of answer processing includes frame and FE alignment to instantiate the vacant FE of the question with its corresponding value in the answer passages. Figure 3.8 shows an example question and its answer processing procedure in the FSB model. Different techniques of frame semantic alignment are proposed and evaluated in the FSB model. These techniques will be detailed in Chapter 6.

The fusion process of the answer lists retrieved by each answer processing model is based on either the scores of the answers or their ranks. In either approach, answer redundancy results in boosting the position of a candidate answer. The fusion module and its different strategies will be discussed in Chapter 8 where the two methods of *score-based* and *rank-based* fusion will be introduced. The default fusion strategy in the experimental QA system is score-based.

When was the comet discovered?

The comet, one of the brightest comets this century, was first spotted by Hale and Bopp, both astronomers in the United States, on July 23, 1995.

semantic parsing →

$BECOMING\_AWARE(TIME)$: When was $BECOMING\_AWARE(PHENOMENON)$: the comet **discovered**?

$BECOMING\_AWARE(PHENOMENON)$: The comet, one of the brightest comets this century, was first **spotted** $BECOMING\_AWARE(COGNIZER)$: by Hale and Bopp, both astronomers in the United States, $BECOMING\_AWARE(TIME)$: on July 23, 1995.

	Question	Passage
Frame alignment	BECOMING_AWARE	BECOMING_AWARE
FE alignment		
Time	*When*	*on July 23, 1995*
Theme	*the comet*	*the comet*
Cognizer	*N/A*	*by Hale and Bopp...*

Figure 3.8: General scheme of frame semantic-based answer identification

3.2.4 Baseline Shallow Semantic Parser

The SHALMANESER shallow semantic parser (Erk and Pado 2006) is used to automatically assign semantic classes - frames in our experiments - and semantic roles - FEs - to questions and passages. SHALMANESER employs supervised learning classifiers in order to disambiguate word senses which correspond to semantic classes (the FRED classifier) and assign semantic roles (the ROSY classifier). SHALMANESER is used not only because it is a state-of-the-art parser; but, because the experiments will be based on an existing well-structured fully automated parser that is trained on the different FrameNet releases for English. The training dataset for SHALMANESER contains more than 133,000 annotated BNC (British National Corpus) examples related to more than 5,700 predicates (Erk and Pado 2006). SHALMANESER accepts plain text, FrameNet XML, TIGER XML (Mengel and Lezius 2000), and SALSA/TIGER XML (Erk and Pado 2004) formats as the input and generates SALSA/TIGER XML outputs. The SALSA/TIGER XML format is an extension of TIGER XML in which the syntax of the text is represented as directed graphs. As SHALMANESER is a loosely coupled tool chain, it can employ different tools at each processing step. Table 3.4 shows the setting that is used in our experiments.

We exploit both versions of SHALMANESER, 1.0 and 1.1, respectively trained on the FrameNet 1.2 dataset and the FrameNet 1.3 dataset to annotate the two question sets of TREC 2004 and TREC 2006 and their corresponding passages.

Table 3.4: SHALMANESER settings at each processing step

Processing step	System	Version
POS-tagging	TNT	2
Lemmatization	TreeTagger	-
Syntactic parsing	Collins' Parser	1.0
Machine learning	Mallet	mallet 0.4

3.2.5 Manual Annotation Tool

To manually correct the automated outputs of SHALMANESER and produce a gold standard annotation based on the TREC 2004 dataset, the SALSA Annotation Tool (SALTO) (Burchardt et al. 2006) is used in this work[7]. It is a graphical user interface for manual shallow semantic annotation. The main advantage of using SALTO is its compatibility with the SHALMANESER output formats. We import the SHALMANESER annotation files into SALTO and manually correct the annotations with a procedure which will be explained in detail in Chapter 5.

[7] Details on why and how manual corrections are carried out can be found in Chapter 5.

3.2.6 Baseline QA Systems

There are two types of baseline QA systems considered in this thesis. The first type contains other existing factoid QA systems and the second type includes specific runs of our implemented QA system.

In terms of other existing QA systems, we focus on the TREC 2004 participant QA systems. Particularly, the 10 best-performing factoid runs in TREC 2004, including LCC's factoid QA runs, are considered. These are used to see how our methods perform on the TREC 2004 dataset relative to the actual TREC 2004 QA systems.

For the second type of baseline runs, we set our implemented QA system with the entity-based model of answer processing. With no frame semantics involved, the entity-based QA runs are considered as baseline runs which are to be enhanced with frame semantic-based model. This is used in both the TREC 2004 and TREC 2006 experiments.

3.2.7 Evaluation Metric

In order to evaluate different answer processing runs in our experimental studies, the mrr measure is used as shown in Equation 3.4 (for this problem ar_i indicates the rank of the first correct answer in a list of answers returned for the question q_i).

For TREC evaluations, systems could previously return 5 answers; however, systems now can only return a single answer. The experiments conducted in this thesis conform to this type of mrr evaluation by returning a single answer per question.

With the filtering process performed on the question sets (described in section 3.2.2) there are no questions with NIL answers in our experiments. Therefore, the NIL precision and NIL recall measures are not applicable and not reported in this thesis.

We implement an automated TREC-friendly answer evaluator that matches the answer strings with the TREC-reported regular expressions of correct answers. Both lenient and strict evaluations are considered. A retrieved answer is scored 0 (not an answer candidate) or 1 (an answer candidate) using our answer evaluator. In scoring an answer we perform a pattern matching process. If the answer string *starts* with a correct answer, it is accepted and scored 1. This is similar to considering regular expressions for pattern matching. The main reason for this type of string matching is to encourage and not apply demerit points on retrieving full answers such as "X, who is a CEO of Y" instead of "X" in response to a PERSON question, for example.

A significance test is carried out for measuring the statistical significance of the answer processing methods where applicable in the following chapters. For this, the paired t-test is carried out. We calculate an individual mrr measure for each single question in the dataset for each answer processing method. Subsequently, we calculate the paired t-value using the individual mrr measures. Similar to the paired t-tests carried out for passage retrieval methods, we then map the paired t-values to

statistical p-values where any value less than 0.05 will indicate a statistically significant result.

3.3 Summary

The methodology for answering the research problems has been detailed in this chapter. This consists of the general approach through which the tests are conducted and the experimental setup to perform the practical analysis of the problems.

In passage retrieval, the MultiText algorithm has been chosen as the baseline passage retrieval algorithm. A new passage scoring and ranking function will be developed for this algorithm. Linguistic boosting of input query analysis will be based on the best-performing method in a set of experimental methods under consideration. The best-performing method will be semantically boosted using the frame semantics in FrameNet. The evaluation metrics of accuracy, *mrr*, average precision, and average recall are used to assess the effectiveness of the methods.

In answer processing, the baseline QA system that we develop for the experiments has been explained with respect to its different modules. A baseline automated shallow semantic parser - SHALMANESER - has also been introduced which is used to annotate the texts with the FrameNet frames and FEs. To manually correct the automated outputs of SHALMANESER, the SALTO annotation tool has been selected which accepts the outputs of SHALMANESER and produces compatible formats of outputs. The main evaluation metric in answer processing will be *mrr* to be consistent with TREC-based evaluations.

The datasets used for passage retrieval and answer processing are the factoid question sets in the TREC 2004 and TREC 2006 QA tracks. Some filtering processes on the datasets will be applied according to each task.

Chapter 4

Enhancing Answer Passage Retrieval for Question Answering

This chapter focuses on enhancing the effectiveness of answer passage retrieval for factoid QA through two main approaches[1]. First, syntactic information, topical/contextual concepts, and other types of information are exploited to improve a baseline passage retrieval algorithm - MultiText - in its final stage of passage scoring and ranking so that it can more effectively retrieve answer passages. Second, we employ the frame semantics encapsulated in FrameNet at the early stage of input query formulation to overcome surface syntactic mismatches between questions and passages and more effectively retrieve and rank answer passages. This is performed on the best-performing passage retrieval method among a set of experimental methods described in Chapter 3.

4.1 Modifying MultiText

The high performance of the MultiText passage retrieval algorithm, as well as its frequent participation in TREC (Clarke et al. 2000), is the main reason for choosing MultiText to be enhanced in our work. We modify MultiText (see section 3.1.3) in a way that it can retrieve a greater number of answer-containing passages for the questions in the TREC QA track. This is performed by modifying the passage scoring and ranking procedure of MultiText using topical information, term density, passage length (the number of terms in each passage), and limited syntactic information.

4.1.1 Approach

To score the passages retrieved by MultiText for a given query, we build representative feature vectors for both the query and passages. Subsequently, the relatedness of each passage to the query is measured using a function that employs the Cosine similarity function and other parameters which

[1] Parts of the work in this chapter have been published in (Ofoghi, Yearwood, and Ghosh 2006*b*).

will be explained in the following paragraphs. By using the Cosine function, the lengths of feature vectors are normalized and the angle between the vectors is measured. This overcomes the problem with longer texts (passages) tending to have large term frequencies. Improvements in normalization over the standard Euclidean norm used in the Cosine measure have been developed. For example see the work by Singhal, Buckley, and Mitra (1996).

The input questions are processed to construct information retrieval queries in a way similar to that explained in section 3.2.3.1. The procedure includes three main steps of stop-word removal, term stemming, and TREC target reference resolution.

In representing queries and passages, the standard vector space model is used in which the feature vector for the query q_i is constructed as $\bar{q}_i = (q_{i1}, q_{i2}, \ldots, q_{iN})$ where N is the size of the index or term dictionary $T = t_1, t_2, \ldots, t_N$ of the text collection (50 documents per TREC target) and q_{ij} refers to the weight of the term t_j for the query q_i. Respectively, the feature vector of a passage p_i is $\bar{p}_i = (p_{i1}, p_{i2}, \ldots, p_{iN})$.

Since the number of query terms in the question sets is not large, we modify this model by taking into consideration only the terms which are present in each query. Therefore, the feature vectors are not of the same size N over the query or question set as the lengths of the vectors vary according to the number of query terms. This makes the computational part more efficient, although it is still totally consistent with the concepts of the standard vector space model.

The weighting scheme for the features in the passage feature vectors and query feature vectors are different. We consider each q_{ij} to be equal to 1 which translates into $\bar{q}_i = (1, 1, \ldots, 1)$. In our experiments, we retrieve $Rank$ passages ($Rank \in \{10, 15, 20\}$) for each question. As a result, the vector set of the passages retrieved for the query q_i is $P_i = \{\bar{p}_{i1}, \bar{p}_{i2}, \ldots, \bar{p}_{iRank}\}$. The vector for each passage p_j retrieved in response to q_i is represented as $\bar{p}_{ij} = (p_{ij1}, p_{ij2}, \ldots, p_{ijn_i})$ where n_i is the size of the query vector $\bar{q}_i$. Equation 4.1 shows the weighting scheme for the features in the passage vectors for the query q_i, where p_{ijk} refers to the kth feature value for the jth passage in the list of retrieved passages, tf_{ijk} is the raw term frequency of the query term t_k in the jth passage for the same query, pl_j is the length of the passage p_j in terms of the number of the actual terms in the passage p_j (which emphasizes short retrieved passages as in the original MultiText algorithm), and $|questions|$ is the total number of questions under experiment. We use $log(pl_j)$ because our trials with pl_j and $log(pl_j)$ indicated less sensitivity to the length of passages and improved performance with $log(pl_j)$. The element w_k is the weight of the term t_k. For efficiency reasons, we do not use IDF-like term weights like those in original MultiText; instead, we consider the following rules to calculate term weights:

- **Rule 1:** The parts-of-speech of the terms are considered. Verbs and adjectives have higher weights (0.8) than nouns, adverbs, and others (0.4). The motivation for emphasizing verbs and adjectives comes from our practical experiments which have shown the importance of verbs and adjectives in more effective passage retrieval. We have carried out basic tests on the

effect of different weights of different part-of-speech terms on the retrieval effectiveness of the modified MultiText algorithm on 60 factoid questions in the TREC 2004 QA dataset. This has confirmed the high influence of verbs and adjectives on the retrieval effectiveness of answers at the level of the top 10 passages. However, these measures (term weights) could possibly be improved by a more sophisticated optimization or machine learning process.

- **Rule 2:** The idea of this rule is to add emphasis to the terms that appear in the TREC targets/topics (see section 1.2 for the definition of TREC targets). To this end, the appearance of the terms in the TREC target of the query (question) is checked. The terms which occur in TREC targets get higher weights. This rule elevates the weights of target-appearing terms, already assigned according to their part-of-speech, to the maximum value of 1.0. If the term t_k is not appearing in the TREC target, then its weight w_k is just assigned based on Rule 1.

$$\begin{aligned} & p_{ijk} = \frac{tf_{ijk}}{log(pl_j) + tf_{ijk}} \times w_k \\ & i = 1 \ldots |questions| \\ & j = 1 \ldots Rank \;\; (where\ Rank \in \{10, 15, 20\}) \\ & k = 1 \ldots n_i \end{aligned} \tag{4.1}$$

Having the feature vectors of the queries and passages established, the relatedness score of a passage to a given query is calculated using the formula that is shown in Equation 4.2.

$$r(q_i, p_{ij}) = cos(\bar{q}_i, \bar{p}_{ij}) \times \frac{c_{ij}}{n_i} \tag{4.2}$$

In Equation 4.2, c_{ij} is the number of the query terms in the query q_i which are covered by the passage p_j and n_i refers to the total number of the query terms in the query q_i. The usage of the factor $\frac{c_{ij}}{n_i}$ is motivated by the original MultiText algorithm where the concept of covering more query terms is emphasized. As a result, passages covering a greater number of query terms will tend to get higher scores in our passage scoring procedure.

In summary, the weighting scheme that we apply to the passage feature vectors in Equation 4.1 and also the calculations in the relatedness function 4.2 carry combinations of different types of information:

- Traditional density-based information of query terms encapsulated in the term frequency tf_{ijk} which is used to emphasize the passages which have greater numbers of query terms. This could be contributed to by any of the query terms and as such, is different from the coverage concept (see below).
- Limited linguistic information at the level of syntax applied by Rule 1 to weight the terms according to their parts-of-speech, to accentuate verbs and adjectives which have shown greater influence in retrieval.
- Topical information enforced by Rule 2 which emphasizes the overall relatedness of the passages

to the queries in terms of the general topic around which the query is centred.

- The length of passages expressed as $log(pl_j)$ which normalizes the feature *values* for the length of the passages. This is a primary level of feature normalization applied before measuring the passage-query similarities using the Cosine similarity function that normalizes the feature *vectors* to the Euclidean length of the vectors (texts).
- The coverage concept borrowed from the original MultiText algorithm represented by $\frac{c_{ij}}{n_i}$ which emphasizes the passages that contain greater numbers of query terms, regardless of their frequency of occurrence.

We use the algorithm explained in (Clarke, Cormack, and Tudhope 2000) in implementing the basics of the MultiText algorithm except for the passage scoring function. The MultiText passage scoring and ranking function is replaced with our procedure of passage scoring to measure the relatedness of the passages retrieved by MultiText.

4.1.2 Experimental Results

In order to evaluate the modified version of the MultiText algorithm, its effectiveness in retrieving answer passages is compared with that of the original MultiText algorithm. In addition, it is evaluated with respect to a set of other passage retrieval methods - the Lemur passage retrieval methods described in Chapter 3 - to observe the overall standing of this modified version of MultiText with respect to some other existing methods.

Table 4.1: Accuracy of modified MultiText compared with those of MultiText and the Lemur passage retrieval methods on 208 TREC 2004 and 386 TREC 2006 factoid questions

Passage retrieval method	trec04			trec06		
	acc@20	acc@15	acc@10	acc@20	acc@15	acc@10
Lemur-TFIDF	61.53st 75.48ln	58.65st 72.59ln	50.96st 65.86ln	53.62st 68.91ln	48.44st 65.80ln	43.78st 62.69ln
Lemur-OkapiBM25	57.69st 69.71ln	51.44st 65.38ln	42.30st 58.17ln	39.63st 60.36ln	37.04st 57.25ln	31.34st 51.55ln
Lemur-CORI_collection_selection	56.73st 70.67ln	51.44st 66.82ln	47.59st 62.01ln	54.14st 69.17ln	50.51st 67.35ln	45.07st 62.95ln
Lemur-Cosine	59.61st 73.07ln	55.76st 68.75ln	49.03st 62.98ln	52.59st 68.13ln	50.25st 66.58ln	43.78st 60.10ln
Lemur-KL_DivergenceLanguage	61.53st 75.00ln	58.17st 72.11ln	49.03st 64.90ln	54.92st 68.65ln	51.81st 66.32ln	46.89st 61.65ln
Lemur-InQuery_CORI	64.90st 77.40ln	61.05st **75.00ln**	51.92st 65.86ln	**55.69st** **70.20ln**	**52.59st** **67.87ln**	**48.18st** **63.98ln**
MultiText	61.53st 72.59ln	57.69st 69.71ln	52.40st 64.42ln	41.96st 54.40ln	40.93st 52.33ln	37.82st 50.25ln
Modified MultiText	**68.26st** **75.96ln**	**65.38st** 74.03ln	**60.57st** **70.19ln**	51.81st 64.76ln	48.44st 60.88ln	45.59st 58.80ln

Table 4.1, Table 4.2, Table 4.3, and Table 4.4 show the results obtained for each passage retrieval method on two datasets, the TREC 2004 and TREC 2006 factoid question sets. These tables show

Table 4.2: The *mrr* values of modified MultiText compared with those of MultiText and the Lemur passage retrieval methods on 208 TREC 2004 and 386 TREC 2006 factoid questions

Passage retrieval method	trec04			trec06		
	mrr@20	mrr@15	mrr@10	mrr@20	mrr@15	mrr@10
Lemur-TFIDF	0.26st 0.39ln	0.26st 0.38ln	0.25st 0.38ln	0.24st 0.38ln	0.24st 0.37ln	0.24st 0.37ln
Lemur-OkapiBM25	0.19st 0.30ln	0.19st 0.30ln	0.18st 0.29ln	0.15st 0.27ln	0.14st 0.27ln	0.14st 0.27ln
Lemur-CORI_collection_selection	0.25st 0.37ln	0.25st 0.37ln	0.25st 0.37ln	0.26st 0.40ln	0.26st 0.40ln	0.26st 0.39ln
Lemur-Cosine	0.27st 0.38ln	0.27st 0.38ln	0.26st 0.38ln	0.26st **0.40ln**	0.26st **0.40ln**	0.25st 0.39ln
Lemur-KL_DivergenceLanguage	0.28st 0.39ln	0.28st 0.39ln	0.27st 0.38ln	0.27st 0.39ln	0.26st 0.39ln	0.26st 0.39ln
Lemur-InQuery_CORI	0.28st 0.39ln	0.28st 0.39ln	0.27st 0.38ln	0.26st **0.40ln**	0.26st **0.40ln**	0.26st **0.39ln**
MultiText	0.32st 0.41ln	0.32st 0.40ln	0.32st 0.40ln	0.20st 0.29ln	0.20st 0.29ln	0.19st 0.29ln
Modified MultiText	**0.36st** **0.43ln**	**0.35st** **0.43ln**	**0.35st** **0.43ln**	**0.28st** 0.39ln	**0.28st** 0.39ln	**0.28st** 0.39ln

the results for different evaluation metrics including accuracy (acc), *mrr*, average precision (prec), and average recall (rec) respectively. The values ending with the string "ln" represent the measures using the lenient evaluation paradigm. The measures that are obtained in accordance with the strict evaluation procedure end with "st". The bold font is used to show the maximum values in each column.

The results regarding accuracy represent the maximum chance of answering questions where our modified MultiText or the other retrieval methods are used in the passage retrieval phase of factoid QA. The *mrr* results compare the effectiveness (and efficiency) that these different retrieval methods deliver to the answer processing module of the QA pipeline. Higher *mrr* values promise more efficient and effective answer processing. The average precision and average recall results are more considerable in the context of traditional information retrieval processes.

To assess the significance of the evaluations, the paired t-test is conducted between the results obtained by the modified MultiText method and those of the other retrieval methods (see section 3.1.4). Table 4.5 shows the significance test probabilities. The test is only performed at the level of top 10 retrieved passages for readability.

4.1.3 Discussion

By analysing the strict results (the lenient results follow a similar trend), a few aspects can be explained. The results of our experiments indicate that in the TREC 2004 dataset, the modified MultiText algorithm outperforms all of the other methods, especially the baseline MultiText algorithm

Table 4.3: Average precision of modified MultiText compared with those of MultiText and the Lemur passage retrieval methods on 208 TREC 2004 and 386 TREC 2006 factoid questions

Passage retrieval method	trec04			trec06		
	prec@20	prec@15	prec@10	prec@20	prec@15	prec@10
Lemur-TFIDF	0.03st 0.04ln	0.04st **0.06ln**	0.06st **0.08ln**	**0.03st** **0.04ln**	0.03st **0.05ln**	0.04st **0.07ln**
Lemur-OkapiBM25	0.03st 0.04ln	0.03st 0.05ln	0.05st 0.07ln	0.02st 0.03ln	0.02st 0.04ln	0.03st 0.05ln
Lemur-CORI_collection_selection	0.03st 0.04ln	0.04st 0.05ln	0.05st **0.08ln**	**0.03st** **0.04ln**	0.03st **0.05ln**	0.05st **0.07ln**
Lemur-Cosine	0.03st 0.04ln	0.04st **0.06ln**	0.05st **0.08ln**	**0.03st** 0.03ln	0.03st **0.05ln**	0.05st 0.06ln
Lemur-KL_DivergenceLanguage	0.03st **0.05ln**	0.04st **0.06ln**	0.05st **0.08ln**	**0.03st** 0.03ln	0.03st 0.05ln	0.05st 0.07ln
Lemur-InQuery_CORI	0.03st **0.05ln**	0.04st **0.06ln**	0.06st **0.08ln**	**0.03st** **0.04ln**	**0.04st** **0.05ln**	**0.05st** **0.07ln**
MultiText	0.03st 0.04ln	0.04st **0.06ln**	0.06st **0.08ln**	0.02st 0.03ln	0.03st 0.04ln	0.04st 0.05ln
Modified MultiText	**0.04st** **0.05ln**	**0.05st** **0.06ln**	**0.07st** **0.08ln**	0.02st 0.03ln	0.03st 0.04ln	0.05st 0.06ln

with a significant margin regarding almost all evaluation metrics. In the TREC 2006 dataset, however, the performance of the modified version of MultiText is a middle performer relative to the other methods, although its performance is still significantly higher than that of the MultiText algorithm. In terms of accuracy, the methods that the modified MultiText algorithm outperforms in the TREC 2006 dataset include the Lemur-TFIDF, Lemur-OkapiBM25, Lemur-CORI_collection_selection, Lemur-Cosine, and MultiText (in some cases the differences are statistically significant) while the two methods Lemur-KL_DivergenceLanguage and Lemur-InQuery_CORI perform better than our modified MultiText.

With respect to *mrr*, our modified MultiText performs best among the set of methods in both datasets. With regard to the average precision and average recall values, again our method shows a middle-level performance in the set of methods.

The fact that our modified MultiText, with a different passage scoring and ranking function, performs significantly better than the original MultiText algorithm with respect to all evaluation metrics (except for *mrr* on the TREC 2004 dataset), ensures that the overall effect of using the different types of information - mentioned in section 4.1.1 - in the scoring and ranking function is positive. Especially in the case of the *mrr* evaluation metric, where the probabilities of the results being chance findings (in the TREC 2006 dataset) are extremely small, the considerable difference can translate into a significantly better overall QA performance as the task of answer extraction and scoring is very much dependent on the number and rank of the answer-bearing passages in the retrieved list of passages per question. In the TREC 2004 dataset, however, the *mrr* values of the MultiText algorithm are not significantly improved by our modified version of MultiText.

Table 4.4: Average recall of modified MultiText compared with those of MultiText and the Lemur passage retrieval methods on 208 TREC 2004 and 386 TREC 2006 factoid questions

Passage retrieval method	trec04			trec06		
	rec@20	rec@15	rec@10	rec@20	rec@15	rec@10
Lemur-TFIDF	0.53st 0.66ln	0.50st 0.63ln	0.45st 0.57ln	0.46st 0.61ln	0.42st 0.58ln	0.38st 0.55ln
Lemur-OkapiBM25	0.50st 0.61ln	0.45st 0.57ln	0.37st 0.50ln	0.34st 0.54ln	0.31st 0.51ln	0.27st 0.45ln
Lemur-CORI_collection_selection	0.49st 0.61ln	0.45st 0.57ln	0.41st 0.54ln	0.47st 0.62ln	0.44st **0.60ln**	0.38st **0.56ln**
Lemur-Cosine	0.52st 0.64ln	0.48st 0.60ln	0.43st 0.54ln	0.46st 0.61ln	0.43st 0.59ln	0.37st 0.53ln
Lemur-KL_DivergenceLanguage	0.53st 0.66ln	0.50st 0.62ln	0.44st 0.56ln	0.48st 0.61ln	0.45st 0.59ln	0.40st 0.54ln
Lemur-InQuery_CORI	0.56st **0.68ln**	0.53st 0.65ln	0.46st 0.57ln	**0.49st** **0.63ln**	**0.46st** **0.60ln**	**0.41st** **0.56ln**
MultiText	0.53st 0.65ln	0.49st 0.61ln	0.44st 0.55ln	0.36st 0.47ln	0.35st 0.45ln	0.32st 0.43ln
Modified MultiText	**0.60st** **0.68ln**	**0.58st** **0.66ln**	**0.53st** **0.61ln**	0.45st 0.57ln	0.42st 0.54ln	0.39st 0.52ln

4.2 Frame Semantic-Based Retrieval Boosting

In boosting the effectiveness of passage retrieval, the definition of quality and effect must be made clear. Some existing works interpret the quality of retrieval to be the number of retrieved consecutive passages from certain documents (Harabagiu et al. 2000; Moldovan et al. 1999). In our work, the boosting procedure focuses on the effectiveness of retrieval where the quality is measured based on the similarity scores of retrieved passages to the queries (a high similarity score for a retrieved passage is interpreted as high quality for that passage).

Most of the existing passage retrieval algorithms are dependent on the occurrences of exact matches of surface features in the queries and the textual documents. As a result, even their state-of-the-art precision of retrieval cannot reach very high levels due to limitations imposed by syntactic structures. Example 4.1 shows a case where surface structures fail to resolve the connection between the answer-bearing passage and the question. The predicate "discover" appears in the question whereas in the answer-containing passage the alternative predicate "spot" is mentioned.

Example 4.1-

> Who discovered Hale-Bopp?
>
> The comet, one of the brightest comets this century, was first spotted by Hale and Bopp, both astronomers in the United States, on July 23, 1995.

These types of mismatches are tackled by other passage retrieval methods which incorporate linguistic information (see Chapter 2). However, there are other types of mismatches which are

Table 4.5: Probabilities (p-values after paired t-tests@10) obtained in the significance test between the results of modified MultiText, MultiText, and the Lemur passage retrieval methods on 208 TREC 2004 and 386 TREC 2006 factoid questions - first and second rows correspond to strict and lenient evaluations respectively - values with † are statistically significant ($p < 0.05$)

Passage retrieval method	trec04				trec06			
	mrr	prec	rec	acc	mrr	prec	rec	acc
Lemur-TFIDF	<0.001†	0.027†	0.006†	0.002†	0.013†	0.396	0.229	0.181
	0.076	0.297	0.090	0.042†	0.236	0.035†	0.051	0.021†
Lemur-OkapiBM25	<0.001†	<0.001†	<0.001†	<0.001†	<0.001†	<0.001†	<0.001†	<0.001†
	<0.001†	0.020†	0.002†	<0.001†	<0.001†	<0.001†	0.002†	<0.001†
Lemur-CORI_collection_selection	<0.001†	0.009†	<0.001†	<0.001†	0.152	0.397	0.323	0.401
	0.035†	0.229	0.011†	0.002†	0.392	0.020†	0.032†	0.015†
Lemur-Cosine	0.002†	0.010†	0.002†	<0.001†	0.103	0.427	0.174	0.164
	0.071	0.215	0.016†	0.005†	0.385	0.127	0.285	0.258
Lemur-KL_DivergenceLanguage	0.006†	0.015†	0.003†	<0.001†	0.186	0.106	0.322	0.254
	0.092	0.336	0.040†	0.028†	0.454	0.035†	0.081	0.073
Lemur-InQuery_CORI	0.005†	0.060	0.015†	0.003†	0.182	0.072	0.132	0.087
	0.083	0.440	0.103	0.053	0.347	0.012†	0.016†	0.005†
MultiText	0.146	0.036†	0.007†	0.005†	<0.001†	<0.001†	<0.001†	<0.001†
	0.212	0.119	0.043†	0.020†	<0.001†	<0.001†	<0.001†	<0.001†

more complicated. In Example 4.2 the paraphrasing instance is harder to resolve as there is no direct relation between the terms "mother" and "son" appearing in the question and the passage.

Example 4.2-

> Who is his [Horus's] mother?
>
> Osiris, the god of the underworld, his wife, Isis, the goddess of fertility, and their son, Horus, were worshiped by ancient Egyptians.

The only clue which connects the two text snippets in Example 4.2 is the general semantics encapsulated in the semantic frame "Kinship" in FrameNet. It is obvious that resolution of such mismatches requires deep semantic analysis of the texts. Such scenario-based relations have not been studied much in this context, especially with respect to the initial step of query analysis for retrieving the most specific passages to a given question.

We try to solve these types of query and passage mismatches by using the frame semantics encapsulated in FrameNet via an iterative and semantic input query analysis step which will be explained in the next section.

4.2.1 Approach

The generalization over conceptual scenarios and their related properties is a major characteristic of FrameNet. We consider this for resolving the problem of poor passage retrieval performance in

the context of QA that occurs as a result of surface mismatches between the terms in the document collection and the query keywords. The semantic generalization applied by FrameNet plays the role of the lost chain for retrieving semantically related passages in response to the queries.

In the context of QA, not all types of semantic query alternatives by rewriting terms are of interest due to the fact that a QA system has to be capable of answering questions with exact answers. For instance, in some cases it is not useful to change the original query using the WordNet semantic relations hypernymy/hyponymy, although this performs well for other information retrieval-based applications such as document retrieval (Voorhees 1994). It may cause the retrieval of more indirectly related passages to the question leading to the extraction of answers which may not be suitable or meaningful due to an undesired generalization/specialization. For instance, if the query "Beth go Paris" for the question "When did Beth go to Paris?" is rewritten in the form of "Beth go city" by generalizing "Paris" to "city", then retrieved passages may not even contain any information regarding "Paris". However, this does not include systems which try to identify online relations between concepts of different abstraction levels (Moldovan et al. 2002) that may result in a beneficial semantic matching of questions and already retrieved passages. The use of synonymy relations also cannot overcome the problem of scenario-based relations like that between the pair "sender-receiver" (also see section 2.1.4). We argue that the methods based on such relations are not suitable for answering direct factoid questions, although they perform well in other contexts.

In addition, query expansion by adding new terms is another way to overcome mismatch problems as fully explained in section 2.1.3. However, query expansion leads to higher recall and lower precision that may not be suitable in a QA framework where high precision is more desired (Pradhan et al. 2002). Our trial experiments show that the expansion process can reduce the level of specificity of the passages and result in retrieval of a greater number of less-relevant passages to the query. From an information retrieval point of view, it simply contributes to recall and damages precision which is more desired in terms of QA (also see section 2.1.4).

In what is called *scenario-based normalization*, the actual procedure of our proposed idea contains a joint generalization-specialization action to provide alternatives for the query terms (the main predicates) which evoke a FrameNet frame. The procedure considers one of the related terms that is inherited from that frame. This generalization-specialization method guarantees that the query remains at the same semantic abstraction level of the original question. For example, when considering the query "Jack son", the keyword "son" evokes a general scenario of "Kinship" which is then specialized to one of the other items covered by the scenario like "father".

These types of passages either cannot be retrieved or have a very low similarity measure with the original query (due to surface mismatches). However, the retrieval performance may be boosted by substituting the target word of the question with semantically related ones in FrameNet. Figure 4.1 shows the cycle of semantically boosting the passage retrieval effectiveness via question rewriting where *original*, *current*, and *alternative* refer to the original query term, the current query term,

and the alternative query term. POS_{org} and POS_{lu} also indicate the part-of-speech of the original query term and that of the Lexical Unit (LU) under consideration respectively.

The process starts with evoking the appropriate FrameNet frame from which the main target predicate of the question inherits and retrieving the passages for the original query. The frame evocation task is performed on the input question where the contextual information helps in evocation of the right frame in terms of the predicate sense. The query formation process is similar to that mentioned in section 3.2.3.1. As long as there is an unseen LU with the same part-of-speech as the frame-evoking target term in that frame, the query is rewritten by substituting the target word with the unseen LU from FrameNet and new passages are retrieved for the rewritten query. In order to

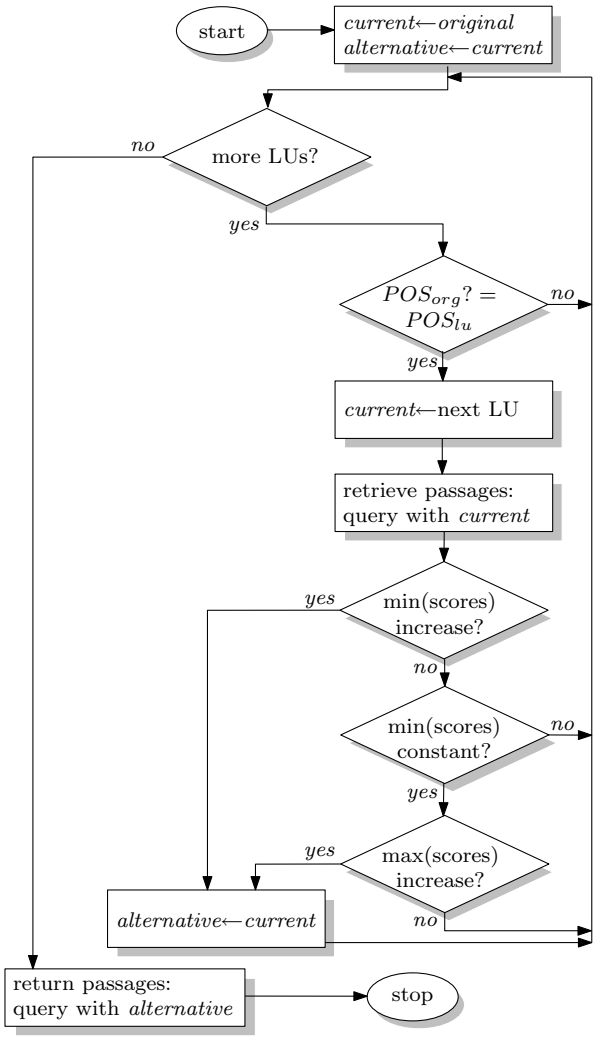

Figure 4.1: Semantic boosting cycle of passage retrieval effectiveness

decide whether the term substitution process has a positive effect on the retrieval, a score analysis of the passages is performed. There are four possible cases when analysing the scores of the top-ranked passages. We formulate our arguments in the following scenarios:

- **Scenario 1-** the minimum score (the score of the nth passage in an ordered list of n retrieved passages) increases: this indicates that the *general relevance* of the top passages to the query rises so that even the least score in the top-ranked passages increases.
- **Scenario 2-** the maximum score increases: such a situation occurs when specificity of the (first) passages increases so that the *maximum relevance* of the passages rises.
- **Scenario 3-** the centroid of the scores increases: this situation alone does not imply any change in the relevance of the passages; however, in conjunction with other scenarios may indicate minor changes of the relevance of the passages.
- **Scenario 4-** the variance of the scores changes: if this happens, it is due to the changes in the lower and upper bounds of the scores. These were discussed in Scenario 1 and Scenario 2.

In accordance with these scenarios, in our semantic boosting cycle, we take into consideration the change of the lower and upper bounds of the passage scores with emphasis on the lower bound. This is because the change in the lower bound relates to the general relevance of the top-ranked passages. When it increases, there is a bigger chance of having the answer-bearing passage(s) included in the list of top-ranked passages due to the shift in the lower scores. In addition, when the general relevance does not change, the maximum relevance - the upper bound or the maximum score - is the second choice for indicating that the answer-containing passage has risen in the list of the top passages. We consider two examples to help explain how our semantic boosting cycle is applied to natural language questions.

Example 4.3-

The question "Who beat him to take the title away?" (Q18.5 in the TREC 2004 QA track) is submitted to the passage retrieval module. This question is formulated into the query "beat boxer Floyd Patterson take title away" after question analysis and TREC target reference resolution. The retrieval module returns a list of passages to this query. As the main predicate "beat.v" evokes the frame "Cause-harm" in FrameNet, the list of alternative terms for this target contains all of the LUs matching the part-of-speech of the original target term such as "bash.v" and "batter.v".

By applying the boosting cycle of Figure 4.1 to the original query (in Figure 4.2) with the list of alternative terms, different sets of passages are retrieved per intermediate reformulated query. When the stopping criterion of the procedure is met, the best alternative term and its corresponding query are selected. In this case, the term "knock" is selected which forms the query "knock boxer Floyd Patterson take title away". The passage retrieval task is then completed by choosing the list of the retrieved passages for this reformulated query. The overall schematic view of the procedure in this example is shown in Figure 4.2.

```
question: "Who beat him to take the title away?" (TREC target:
boxer Floyd Patterson) →
query: "beat boxer Floyd Patterson take title away" →
main predicate "beat" evokes the frame "Cause-harm" →
part-of-speech checking on LUs →
iterative query rewriting using alternative predicates →
iterative passage retrieval for reformulated queries →
best alternative selection →
term: "knock" →
fetch passages retrieved for the query: "knock boxer Floyd
Patterson take title away" →
stop.
```

Figure 4.2: Schematic view of Example 4.3

While the original query "beat boxer Floyd Patterson take title away" did not evoke the answer-bearing passage, the alternative query "knock boxer Floyd Patterson take title away" containing the LU "knock.v" instead of "beat.v" does effectively manage the retrieval of the passage which contains the actual correct answer to the question "Who beat him to take the title away?". This alternative is shown to result in the highest passage scores among all other possible terms in the frame "Cause-harm". This ensures that the score analysis procedure is able to pick up the best alternative LU effectively.

Example 4.4-

The next question "Who was his mother?" (Q14.3 in the TREC 2004 QA track) has the target string "Horus". The schematic view of the process is shown in Figure 4.3. In this example, there is no direct relation between the terms "mother" and "son"; however, using the encapsulated frame semantics in FrameNet it is possible to substitute these terms with each other and bring the answer-bearing passages up in the list of the top-ranked passages. In this specific example, because of the fact that there are only two keywords in the query, it is more crucial to substitute the original term with its best alternative; otherwise, the retrieved passages can be much farther from the desired specifically related passages.

4.2.2 Experimental Results

In order to evaluate the effectiveness of the proposed semantic boosting mechanism of the passage retrieval task for QA, we conduct a number of runs in our experimental setting explained in Chapter 3. Semantic boosting is applied over the best-performing algorithm to observe any improvement that can be achieved on the upper bound of the effectiveness of answer passage retrieval by this method. Table 4.6 shows the methods over which the boosting cycle is performed in the two datasets. The

```
question: "Who was his mother?" (TREC target: Horus) →
query: "Horus mother" →
main predicate "mother" evokes the frame "Kinship" →
part-of-speech checking on LUs →
iterative query rewriting using alternative predicates →
iterative passage retrieval for reformulated queries →
best alternative selection →
term: "son" →
fetch passages retrieved for the query: "Horus son"→
stop.
```

Figure 4.3: Schematic view of Example 4.4

main criterion used to select the methods to be semantically boosted is the accuracy rate of the passage retrieval methods reported in section 4.1.2 for the two datasets.

Table 4.6: The methods selected for semantic boosting

trec04		trec06	
strict	lenient	strict	lenient
Modified MultiText	Modified MultiText	Lemur-InQuery_CORI	Lemur-InQuery_CORI

Table 4.7: Accuracy analysis of semantic boosting on 208 TREC 2004 and 386 TREC 2006 factoid questions

Passage retrieval method	trec04			trec06		
	acc@20	acc@15	acc@10	acc@20	acc@15	acc@10
Non-boosted method	68.26st 75.96ln	65.38st 74.03ln	60.57st 70.19ln	55.69st 70.20ln	52.59st 67.87ln	48.18st 63.98ln
Boosted method	**69.71st** **76.92ln**	**66.82st** **75.48ln**	**62.50st** **72.11ln**	**56.21st** **70.72ln**	**52.84st** **68.13ln**	**48.44st** **64.24ln**

To initiate the semantic boosting cycle it is essential to annotate the questions with FrameNet frames. Their FEs are not necessary in this task. We use the SHALMANESER shallow semantic parser introduced in Chapter 3 which is trained with the FrameNet 1.2 data. This parser can evoke frames and assign their FEs, although we only consider the frames on this occasion. The automated outputs of SHALMANESER[2], being incomplete from a human's point of view, are, however, manually corrected to have all possible frames considered with respect to the FrameNet 1.3 data. This is done so that the hypothesis being tested here is not adversely affected by deficiency in SHALMANESER.

[2]The annotation performance of SHALMANESER will be analysed in Chapter 5.

Table 4.8: *mrr* analysis of semantic boosting on 208 TREC 2004 and 386 TREC 2006 factoid questions

Passage retrieval method	trec04			trec06		
	mrr@20	mrr@15	mrr@10	mrr@20	mrr@15	mrr@10
Non-boosted method	0.36st 0.43ln	0.35st 0.43ln	0.35st 0.43ln	**0.26st** **0.40ln**	**0.26st** **0.40ln**	**0.26st** **0.39ln**
Boosted method	**0.37st** **0.44ln**	**0.37st** **0.44ln**	**0.37st** **0.44ln**	**0.26st** **0.40ln**	**0.26st** **0.40ln**	**0.26st** **0.39ln**

Table 4.9: Average precision analysis of semantic boosting on 208 TREC 2004 and 386 TREC 2006 factoid questions

Passage retrieval method	trec04			trec06		
	prec@20	prec@15	prec@10	prec@20	prec@15	prec@10
Non-boosted method	**0.04st** **0.05ln**	**0.05st** **0.06ln**	**0.07st** **0.08ln**	**0.03st** **0.04ln**	**0.04st** **0.05ln**	**0.05st** **0.07ln**
Boosted method	**0.04st** **0.05ln**	**0.05st** **0.06ln**	**0.07st** **0.08ln**	**0.03st** **0.04ln**	**0.04st** **0.05ln**	**0.05st** **0.07ln**

The results of running the semantically boosted methods and their non-semantically boosted versions are shown in Table 4.7, Table 4.8, Table 4.9, and Table 4.10 representing accuracy, *mrr*, average precision, and average recall respectively.

The statistical significance of the differences between the evaluation measures of the non-boosted method and its boosted version is shown in Table 4.11. This table contains the results of the paired t-tests at the level of top 10 passages per question (see section 3.1.4) for the evaluation metrics under consideration.

4.2.3 Discussion

As can be seen in the tables in section 4.2.2, the semantically boosted method performs slightly better than the non-boosted method in most of the TREC 2004 and TREC 2006 experiments. Therefore, the frame semantic-based input query analysis shows some potential to improve the upper bounds of effectiveness of answer passage retrieval for QA across the different metrics except for average precision.

By exploiting the scenario-based relations between the LUs in FrameNet frames and resolving the surface mismatches between a given question and answer-containing passages, the overall improvement over the non-boosted method is achieved in two ways:

i) There are more questions for which the semantically boosted passage retrieval method finds the answers at the certain level of top-ranked passages. For these questions, the answer-bearing passages were not retrieved previously without the semantic boosting cycle applied.

Table 4.10: Average recall analysis of semantic boosting on 208 TREC 2004 and 386 TREC 2006 factoid questions

Passage retrieval method	trec04			trec06		
	rec@20	rec@15	rec@10	rec@20	rec@15	rec@10
Non-boosted method	0.60st	0.58st	0.53st	**0.49st**	**0.46st**	0.41st
	0.68ln	0.66ln	0.61ln	**0.63ln**	**0.60ln**	**0.56ln**
Boosted method	**0.62st**	**0.59st**	**0.55st**	**0.49st**	**0.46st**	**0.42st**
	0.69ln	**0.67ln**	**0.62ln**	**0.63ln**	**0.60ln**	**0.56ln**

Table 4.11: Probabilities (p-values after paired t-tests@10) obtained in the significance test between the results of the non-boosted method and its semantically boosted version on 208 TREC 2004 and 386 TREC 2006 factoid questions - first and second rows correspond to strict and lenient evaluations respectively - values with † are statistically significant ($p < 0.05$)

trec04				trec06			
mrr	prec	rec	acc	mrr	prec	rec	acc
0.301	0.324	0.277	0.217	0.470	0.453	0.467	0.426
0.312	0.340	0.254	0.173	0.478	0.450	0.466	0.414

ii) There is a ranking increase for the answer-bearing passages in some cases. This results only in a *mrr* increase for the retrieval method yielding more effective and efficient consequent answering process in a QA system.

The overall improvement achieved by the boosted retrieval method, however, is not statistically significant, as shown in Table 4.11. When conducting the experiments and a basic error analysis, we found that there would be further possibility for improvement in the retrieval task using the semantically boosted method if there were more predicates covered in FrameNet. The coverage of predicates in FrameNet is an ongoing challenge taken on by the developer group[3]. This suggests better performance for the frame semantic-based boosting cycle.

4.3 Summary

The MultiText passage retrieval algorithm has been enhanced with a new passage scoring and ranking function which uses different types of information. The limited syntactic information - the part-of-speech - of the query terms, the density-based information of the terms, the topical focus of the queries, the length of the retrieved passages, and the rate of covering the query terms by each passage have been considered and shown to improve the effectiveness of the MultiText algorithm in retrieving answer passages.

[3] We will analyse the FrameNet lexical coverage in Chapter 7.

For further enhancement of answer passage retrieval, a frame semantic-based boosting method has been proposed and evaluated which further increases effectiveness in retrieving answer passages. The boosting cycle is applied at the early stage of input query analysis to overcome surface mismatches between queries and answer passages by selecting the best query formulation. The method improves the upper bound of retrieval effectiveness slightly. This is promising given the current state of incomplete coverage of FrameNet over predicates.

Chapter 5

The Effect of Levels of Frame Semantic Parsing on Answer Processing

To analyse the impact of different levels of frame semantic parsing using FrameNet on factoid answer processing, the fully automatically annotated outputs of a baseline shallow semantic parser are manually corrected and different levels of annotation are used. Annotation is performed on the answer passages and question sets to enable frame semantic alignment for the task of answer candidate identification and scoring. The levels of parsing are based on levels of frame evocation, FE assignation, and the part-of-speech of frame evoking elements (FEEs) ranging from the automated labeling instances with limited part-of-speech FEEs to full human level annotations. In this chapter, the contributions of different levels of frame semantic annotation, with respect to individual subtasks of frame evocation and FE assignment, will be measured[1]. The overall effect of more sophisticated annotation (the overall conjunction of frame and FE assignment) on the performance of FrameNet-based factoid answer processing will also be quantified.

5.1 Related Work

The task of shallow semantic parsing mainly consists of two phases: i) sense disambiguation of the predicative target word to identify the semantic class that it covers, and ii) role assignment to the arguments of the predicate with regard to its specific sense (Erk and Pado 2006).

There has been some work to tackle the problem of frame semantic role labeling formally starting with the work by Gildea and Jurafsky (2002) which introduces the problem as a classification task. This approach is followed in other studies (Erk 2006; Erk and Pado 2005; Erk and Pado 2006; Frank 2004; Giuglea and Moschitti 2006; Honnibal and Hawker 2005; Litkowski 2004; Shi and Mihalcea

[1] Some results of this study have already been published in (Ofoghi, Yearwood, and Ma 2008*b*) and (Ofoghi, Yearwood, and Ma 2009).

2004; Thompson, Levy, and Manning 2003).

The task of semantic class and role labeling by shallow semantic parsers has not usually been exploited in QA. Narayanan and Harabagiu (2004*a*) and Narayanan and Harabagiu (2004*b*) were first to introduce the importance of semantic classes and roles in question answering. Their approach is based on the identification of predicate-argument structures using both the FrameNet and PropBank datasets. Similar methods of answer processing are studied in LCC's CHAUCER QA system in TREC 2006 (Hickl et al. 2006). CHAUCER uses FrameNet frames and FEs as one of the answer processing methods with a straightforward frame and FE alignment procedure between the question and answer-containing sentences annotated with the FrameNet data. They also use a PropBank-based semantic parser to generate natural language predictive questions on the basis of each predicate found in the top-ranked passages per question.

The ASSERT shallow semantic parser (Pradhan et al. 2004) is also used in TREC 2005 (Sun et al. 2005) to add the predicate-argument structure from PropBank to texts. From the analysis in (Sun et al. 2005), PropBank-based semantic annotation did not perform very well in extracting answers for factoid and list questions in the TREC 2005 track using semantic structure matching. This was explained to be due to the parser's low recall (the ratio of correctly assigned items divided by the total number of items assigned in a standard annotated corpus). A robust method of using PropBank-based annotations in (Schlaefer et al. 2007) is demonstrated to achieve a median factoid accuracy 0.131 (with maximum accuracy 0.208) which is not close to the state-of-the-art factoid accuracy. One main reason for low performances of these systems is their sole dependence on verb predicates while other part-of-speech predicates are ignored in semantic matching of questions and answer passages.

The study by Shen and Lapata (2007) formulates the usage of frame semantic role labeling via bipartite graph optimization and matching for answer processing using FrameNet frames and FEs. They exploit a soft semantic role labeling technique and an optimization method to overcome the problem of multiple-labels or no-labels for the semantic roles. They have not, however, studied the two main research questions in this chapter: i) the impact of different levels of annotation on QA, and ii) the impact of the different subtasks of frame identification and FE assignment on the same task individually.

Other works - reviewed in Chapter 2 - which use FrameNet and other linguistic resources such as PropBank are not considered to be directly related to the domain of shallow semantic parsing and factoid answer processing as they attain the task in different ways.

5.2 Levels of Frame Semantic Parsing

The semantic parsing performance of SHALMANESER (see section 3.2.4) is below the performance that a human annotator achieves. The more challenging part of semantic parsing is the task of

semantic role assignment which is achieved by a trained classifier, ROSY. This classifier performs poorly compared to the FRED classifier for semantic class identification (Erk and Pado 2006). We are more interested in the overall performance of SHALMANESER for factoid answer processing. Without judging SHALMANESER, we evaluate the possible contribution of a frame semantic-based answer processing method across different levels of shallow semantic parsing.

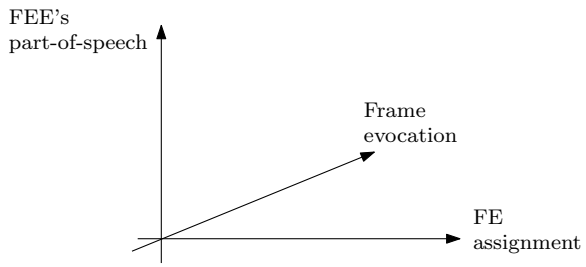

Figure 5.1: The three different facets of FrameNet-based annotation

There are three main concerns when considering different facets of parsing shown in Figure 5.1. The first two tasks of frame evocation and FE assignment are two subtasks of shallow frame semantic parsing. The third, the FEEs' part-of-speech, is considered in an attempt to understand the impact of the different part-of-speech predicates on FrameNet-based factoid answer processing.

With regard to the third aspect - the part-of-speech of the FEEs - two levels of verb FEEs and all FEEs are considered. These are selected to observe the impact of other part-of-speech predicates in factoid answer processing compared to the verb-only scheme available with other linguistic resources such as VerbNet and PropBank. In the experiments, the level of verb-only frames contains the SHALMANESER-evoked verb frames and these have their FEs manually corrected. We, therefore, consider four levels of annotation (parsing) in our study:

- **SHAL:** where frames and their FEs are those evoked by SHALMANESER. There is no manual correction in this level of annotation.
- **SHAL-AF:** where frames are those evoked by SHALMANESER. Their FEs are manually corrected so that there are no wrong or missing assignations with respect to FEs.
- **SHAL-VF:** This is a level where SHAL-AF is reduced to verb-only frames to study the impact of different part-of-speech frames on answer processing performance. This provides an opportunity to compare frame semantic-based answer processing performance with verb-only-based approaches which make use of linguistic resources such as PropBank and VerbNet.
- **SHAL-HL:** Two activities are carried out in this human level annotation at the same time:
    - *First:* There are more frames manually evoked in this level of annotation. The added frames are those from FrameNet frame sets which have not been evoked where required in the texts (we do not add any new frames outside of the FrameNet frame set). The FEs

of these frames are perfectly assigned.

- *Second:* The output of the first step is further analysed manually and the SHALMANESER frame evocations are corrected. Any miss-classification of word senses into wrong frames is rectified. The FEs of all frames are perfectly assigned to the arguments of predicates.

5.3 Two-Step Gold Standard Annotation

In FrameNet the semantic class is realized as the specific frame which is evoked in the true sense of the sentence (Erk 2006), while the roles are the different FEs in that frame. From this point of view, an example of the process of semantic analysis can be shown as in Figure 5.2.

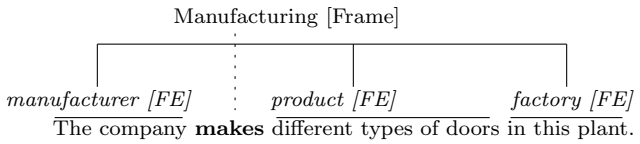

Figure 5.2: Shallow semantic analysis of an example sentence evoking the frame "Manufacturing"

Annotation of the example sentence "the company makes different types of doors in this plant" (in Figure 5.2) with respect to the predicate "make.v" consists of two stages: i) the identification of the right semantic class or frame, and ii) the assignment of the different parts of the sentence (arguments) to the semantic roles or the FEs of the particular frame. There are different semantic classes that can be evoked by the predicate "make.v" such as arriving, building, cooking-creation, causation, and manufacturing. The task of finding the right frame from this set of related semantic classes is a problem which can be formulated as a word sense disambiguation problem (Erk 2006). Having the correct frame identified, semantic role assignment to connect the FEs and the sentence constituents is the next step.

5.3.1 Approach

To have completely annotated answer passages and questions, we address the annotation task in two steps of i) automated shallow semantic parsing, and ii) manual correction of the automated annotations[2]. A description of the SHALMANESER parser for automated shallow frame semantic parsing and the SALTO annotation tool for manual corrections has been given in Chapter 3.

The task of manual correction is an exhaustive process which thoroughly examines each sentence word-by-word. It includes:

- **Frame evocation:** if a predicate could have evoked a correct frame in FrameNet but has not evoked any frame (through SHALMANESER), then the frame is manually invoked and the

[2] The major descriptions of this work have been published in (Ofoghi, Yearwood, and Ma 2007).

FEs are assigned.

- **Frame change:** in case the frame which has already been evoked by SHALMANESER/FRED is not of the correct semantic class of the predicate, the frame is deleted and the right frame is invoked manually. If there is no right frame in FrameNet, due to lack of lexical coverage, then the predicate evokes no frame.
- **FE assignment:** when parts of a sentence (arguments of the predicates) could have been assigned to FEs,; however, have not been assigned to any part of the sentence, then the FEs are assigned to the arguments manually.
- **FE assignment correction:** where there is a need for changing the connectivity of the arguments to the FEs of a frame indicated by SHALMANESER/ROSY, it is performed manually.

Figure 5.3 shows the sentence "Prusiner won a Nobel Prize last year for discovering prions" automatically annotated with the FrameNet elements using SHALMANESER. The annotation output is visualized in SALTO. In the manual correction process of this example, the frame "Finish-competition" is added, the wrongly assigned frame "Duration" is eliminated, and the FEs of the two frames "Calendric-unit" and "Becoming-aware" are corrected in their corresponding sentence segments. Figure 5.4 depicts the annotated sentence after the manual corrections are performed.

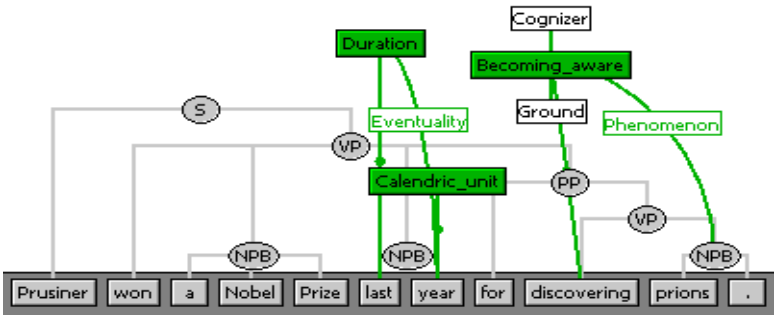

Figure 5.3: Incomplete automated shallow frame semantic parsing of an example sentence by SHALMANESER *before* manual correction

In the manual correction phase, in order to develop the most comprehensive and up-to-date annotation, we use FrameNet 1.3 data with 795 semantic frames, although the SHALMANESER classifiers for this task are trained with FrameNet 1.2 containing 609 semantic frames.

5.3.2 Annotated Corpus

We have annotated a number of TREC 2004 factoid questions and their top 10 answer passages from the AQUAINT text collection. Statistical information about the annotated corpus is summarized in Table 5.1.

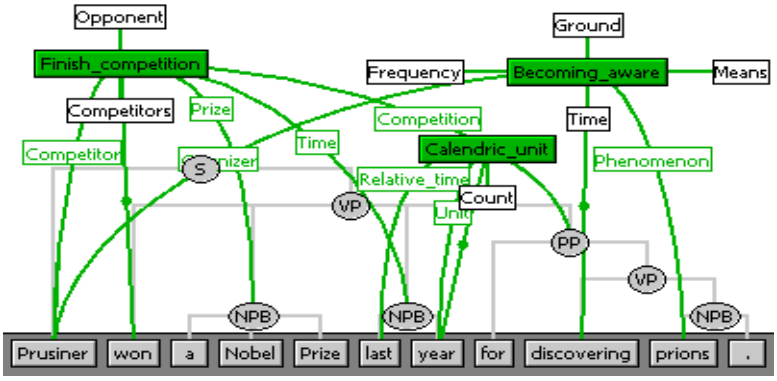

Figure 5.4: Comprehensive frame semantic annotation of an example sentence *after* manual correction

The 1379 passages are extracted in response to the information request of a subset of the TREC 2004 factoid questions including 143 questions for which the retrieval system retrieves passages actually containing the correct answers. The limitation for the task of passage retrieval is set to retrieve the top 10 passages per question. For a few questions, the retrieval system could not retrieve exactly 10 passages (in some instances fewer passages) as there is not enough information text in the collection specifically related to the question. The modified version of the MultiText passage retrieval algorithm - explained in Chapter 4 - is used for this purpose.

Table 5.1: Statistical information of the annotated data

	Passages	trec04 factoid questions
Total	1379	143
Total no. of Sentences	3451	143
Avg. sentences per item	2.502	1.0
Total no. of Terms	89434	864
Avg. terms per sentence	25.915	6.041
Total no. of Terms (unique)	9291	305
Total no. of Predicates	53215	481
Total no. of Predicates (unique)	8121	258

The TREC 2006 dataset has only been annotated using SHALMANESER. The manual correction of these annotations (the TREC 2006 question set and their related passages) was beyond the scope of the thesis.

5.3.3 Statistics of Annotation

The manual correction process includes adding and changing many of the frame and FE assignments. To have a better picture of the task, the two subtasks of the manual correction - frame changes and

FE corrections - are separately analysed statistically as shown in Table 5.2. In Table 5.3 and Table 5.4 the measures are normalized with respect to the number of sentences and terms respectively. The average measures in passages correspond to the sets of 10 passages per question.

With respect to the FrameNet elements - frames and FEs - the statistical measures are summarized in Table 5.5. The total number of unique frames evoked in the answer passage corpus - 592 - covers ∼74% of the total frames in the FrameNet data release 1.3. On the other hand, the overall number of the frames in this corpus - 21741 - represents a rate of 6.299 frames per sentence on average. In the question corpus, the total number of frames drops to 229 with 85 unique frames covering only ∼11% of the FrameNet 1.3 frames. The concurrency rate in this corpus decreases to 1.601 frames per sentence.

Table 5.2: Average number of frames and FEs added/changed in manual correction - not-normalized measures

Parsing level	Passages		trec04 factoid questions	
	Avg. #Frames	Avg. #FEs	Avg. #Frames	Avg. #FEs
SHAL-VF	N/A	19.223	N/A	0.405
SHAL-AF	N/A	35.741	N/A	0.839
SHAL-HL - FN1.2	60.006	158.517	0.524	2.020
SHAL-HL - FN1.3	74.244	182.006	0.566	2.090

Table 5.3: Average number of frames and FEs added/changed in manual correction - normalized by the number of sentences

Parsing level	Passages		trec04 factoid questions	
	Avg. #Frames	Avg. #FEs	Avg. #Frames	Avg. #FEs
SHAL-VF	N/A	0.841	N/A	0.405
SHAL-AF	N/A	1.560	N/A	0.839
SHAL-HL - FN1.2	2.578	6.851	0.524	2.020
SHAL-HL - FN1.3	3.186	7.855	0.566	2.090

Table 5.4: Average number of frames and FEs added/changed in manual correction - normalized by the number of terms

Parsing level	Passages		trec04 factoid questions	
	Avg. #Frames	Avg. #FEs	Avg. #Frames	Avg. #FEs
SHAL-VF	N/A	0.031	N/A	0.075
SHAL-AF	N/A	0.057	N/A	0.132
SHAL-HL - FN1.2	0.096	0.254	0.085	0.325
SHAL-HL - FN1.3	0.118	0.291	0.092	0.336

With this statistical information on the manual correction of the SHALMANESER outputs, the recall and precision of SHALMANESER and SHAL-AF level of annotation are measured. The values

Table 5.5: FrameNet-oriented statistics of the annotated data

	Passages	trec04 factoid questions
#Frames evoked	21741	229
#Frames evoked (unique)	592	85
#FEs assigned	40589	457
#FEs assigned (unique)	2586	202

in Table 5.6 and Table 5.7 are based on the AQUAINT passages and TREC 2004 factoid questions[3]. They consider both the FrameNet 1.2 and FrameNet 1.3 datasets in the standard evaluation level (human level annotation) respectively.

Table 5.6: Parsing evaluations - retrieved passages from the AQUAINT collection

FrameNet dataset at evaluation standard level	Parsing level	Frames			FEs		
		rec	prec	F_1	rec	prec	F_1
FN1.2	SHAL	41.72	73.75	53.29	16.98	43.04	24.35
	SHAL-AF	41.72	73.75	53.29	43.49	100.00	60.62
FN1.3	SHAL	37.97	73.75	50.13	15.64	43.04	22.94
	SHAL-AF	37.97	73.75	50.13	40.01	100.00	57.15

Table 5.7: Parsing evaluations - factoid questions from the TREC 2004 track

FrameNet dataset at evaluation standard level	Parsing level	Frames			FEs		
		rec	prec	F_1	rec	prec	F_1
FN1.2	SHAL	59.20	84.38	69.58	13.17	51.97	21.01
	SHAL-AF	59.20	84.38	69.58	53.41	100.00	69.63
FN1.3	SHAL	57.10	84.38	68.10	12.85	51.97	20.60
	SHAL-AF	57.10	84.38	68.10	52.13	100.00	68.53

The recall and precision values are measured according to the definitions in the literature of parsers evaluation (Carroll, Briscoe, and Sanfilippo 1993). In our specific evaluation process, therefore, the definitions are:

- **Recall:** the ratio of the correct items (frames or FEs) in the parsing level under consideration to the total number of items in the gold standard annotation (SHAL-HL). For instance, if in the SHAL level of parsing there are m correctly evoked frames compared with the frames evoked in the SHAL-HL level where there are n frames evoked in total, then the recall of frame evocation in the SHAL level of parsing is $\frac{m}{n}$.
- **Precision:** the ratio of the correct items in the parsing level under consideration to the total number of items in the parsing level under consideration. For example, if in the SHAL level

[3] The F_1 column represents the standard $F = 2 \times \frac{rec \times prec}{rec + prec}$

of parsing there are m_c correctly evoked frames and m_{ic} frames incorrectly evoked compared with the frames evoked in the SHAL-HL level, then the precision of frame evocation in the SHAL level of parsing is $\frac{m_c}{m_c+m_{ic}}$.

As shown in Table 5.6 and Table 5.7, the recall and precision values at the fully automated shallow semantic parsing level (SHAL) on the open domain texts of the AQUAINT collection are not promisingly high. The task of FE assignment, especially, seems to be a challenging process where precision is not reaching more than 44% in the passage corpus. The recall and precision values for frame-based analyses are equal at SHAL and SHAL-AF levels since there are no frame differences between the two parsing levels. With respect to the FEs, however, the manual corrections of annotations improve recall, precision, and F-measures.

By comparing the corresponding measures for FrameNet 1.2 and 1.3 datasets in both tables, it is observed that the higher lexical coverage of the FrameNet 1.3 dataset results in decreased recall values due to the higher number of items (frames and FEs) in the standard evaluation level of annotation. The precision values, however, remain unchanged; therefore, F-measures decrease.

5.3.4 Quality of Annotation

An important aspect of the annotation task, when human judgments and corrections are included, is the quality of the output annotation with respect to the two main subtasks, namely frame evocation and FE assignment. The manual correction process, in our work, is conducted by a single annotator; however, there is a method for validating the output annotation with respect to the inter-annotator agreement rates.

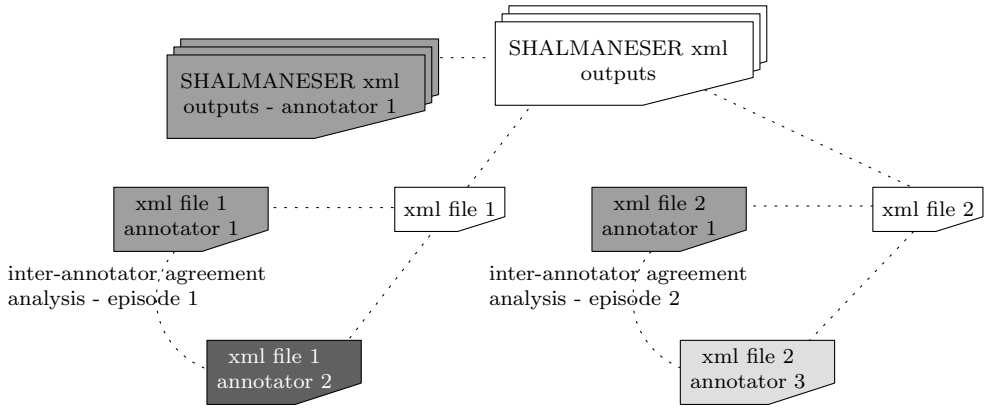

Figure 5.5: The scenario of analysing inter-annotator agreement on the annotated data

After finishing the manual correction task by the sole annotator, two separate portions (10 passages each) of the same SHALMANESER outputs (not the whole set) are annotated by two other

annotators (three annotators in total). Each portion is then augmented by an annotator - portion 1 by annotator2 and portion 2 by annotator3. With this setting, there are two portions annotated by two annotators where the pairs are annotator1-annotator2 and annotator1-annotator3. In two separate episodes, the inter-annotator agreement is measured for frame evocations and for FE assignations. Figure 5.5 shows the scenario. The overall estimated agreement is then calculated as the average values on the two measure sets.

The *alpha* statistic has been used in other similar tasks for frame agreement calculation between annotators (Erk et al. 2003). In this task, we use the *Kappa* statistic (Cohen 1960) as shown in Equation 5.1 where $P(A)$ indicates the observed agreement among the annotators (the probability of the agreed items over the total number of items coded) and $P(E)$ is the expected agreement.

$$\kappa = \frac{P(A) - P(E)}{1 - P(E)} \tag{5.1}$$

The computation of $P(E)$ as the probability of agreement among annotators by chance is the challenging part in the Kappa statistic which can be approached in different ways. We use the Siegel and Castellan's agreement table (Eugenio and Glass 2004) to compute the expected agreement $P(E)$.

$$P(E) = \sum_j \left(\frac{\sum_i n_{ij}}{N \times k}\right)^2 \tag{5.2}$$

Equation 5.2 shows how they calculate the $P(E)$ measure for any number of possible labels, where N is the total number of observations, k is the total number of labels that annotators can assign to each item, and n_{ij} is the number of codings of the label j to the item i. For each predicate in the corpus, we consider four labels:

- No frame (nfr): is used for the predicates that are not assigned to any frame by the annotators.
- Frame by annotator1 (f_{a1}): indicates that a frame has been chosen by annotator1 ($a1$).
- Frame by annotator2 (f_{a2}): indicates that a frame has been chosen by annotator2 ($a2$). This may not be the same as the frame selected by annotator1.
- Frame (fr): is used for the cases where the annotators agree on choosing the same frames.

Table 5.8 shows an example agreement table for 10 predicates $Pred_1, Pred_2, \ldots, Pred_{10}$. For this example agreement table, the Siegel and Castellan's Kappa ($\kappa_{(S\&C)}$) is calculated as follows. First, $P(A)$ is calculated as $\frac{6}{10} = 0.600$ (at 6 rows there are agreements indicated by the number 2). Second, for each label j, p_j - the proportion of predicates assigned to label j - is calculated using the formula in Equation 5.3.

$$p_j = \frac{1}{N \times k} \sum_i n_{ij} \tag{5.3}$$

With $k = 4$ (as there are 4 possible labels for each predicate) and $N = 10$, we have $p_{nfr} = 0.225$, $p_{f_{a1}} = 0.075$, $p_{f_{a2}} = 0.050$, and finally $p_{fr} = 0.150$. Having these values per label, the overall

Table 5.8: An example frame agreement table for 10 predicates with four possible labels assigned by two annotators

Predicate	nfr	f_{a1}	f_{a2}	fr
$Pred_1$	1	1	0	0
$Pred_2$	1	1	0	0
$Pred_3$	1	0	1	0
$Pred_4$	2	0	0	0
$Pred_5$	2	0	0	0
$Pred_6$	2	0	0	0
$Pred_7$	0	1	1	0
$Pred_8$	0	0	0	2
$Pred_9$	0	0	0	2
$Pred_{10}$	0	0	0	2

expected agreement $P(E)$ is equal to 0.079. Finally, the $\kappa_{(S\&C)}$ measure is $\frac{0.600-0.079}{1-0.079} = 0.565$ which shows a middle agreement rate in the range of $[\sim 0, 1]$ on the labeling task performed by the two annotators.

There are different possibilities for measuring the frame evocation agreement with regard to the total number of predicates (N). We calculate the agreement with respect to four predicate counts:

i) All predicates in the corpus (APd),
ii) The maximum of the FEEs labelled by the annotators (Max-FEEA),
iii) Union of the FEEs labelled by the annotators (Un-FEEA), and
iv) All FEEs covered in the FrameNet dataset (FN-FEE).

Table 5.9 summarizes the frame agreement rates obtained with the different predicate counts in two episodes and the average agreement rate which is estimated to be expected in the whole annotated data.

The inter-annotator agreement on the FE assignment task is, however, more problematic because:

- The different annotators may assign slightly different string segments to the same FEs as there is no boundary detection performed to identify and unify the set of arguments in the sentences prior to the manual correction.
- The task of comparison between the FEs assigned by the two annotators is not very well addressed as it is not obvious which FEs need to be aligned.
- The total number of FEs over which the agreement is calculated is not constant. That is, the identification of a baseline set of the FEs to calculate the agreement on is a challenge.

With the above-mentioned challenges, we set different strategies for measuring the agreement rate in the FE assignment subtask. We consider both *exact* and *partial* matches between the instances (arguments) assigned to the FEs[4]. On the other hand, we consider two overall sets of FEs to calculate the agreement over:

[4]Exact and partial matches refer to the situations where the text strings assigned to FEs can be matched with

Table 5.9: Inter-annotator frame agreement rates $\kappa_{(S\&C)}$

Analysis episode	Frame evocation agreement			
	APd	Max-FEEA	Un-FEEA	FN-FEE
annotator1-annotator2	0.804	0.387	0.323	0.661
annotator1-annotator3	0.789	0.378	0.356	0.708
Avg. agreement	0.796	0.382	0.339	0.684

Table 5.10: Inter-annotator FE agreement rates

Analysis episode	Frame evocation agreement			
	exact match		partial match	
	Max-FEA	Un-FEA	Max-FEA	Un-FEA
annotator1-annotator2	17.100	14.420	25.278	21.316
annotator1-annotator3	29.032	31.629	36.363	39.616
Avg. agreement	23.066	23.024	30.820	30.466

- The union set of the FEs assigned by the two annotators (Un-FEA), and
- The maximum set (number) of the FEs assigned by either annotator (Max-FEA).

The method of calculation of the FE agreement is based on the percentage of the agreed FEs over the total number of FEs. Equation 5.4 shows the formula in which $|ag\_fes|$ is the total number of agreed FEs and $|all\_fes|$ indicates the number of all FEs. This set can be either Un-FEC or Max-FEC mentioned above.

$$Agreement_{fes} = \frac{|ag\_fes|}{|all\_fes|} \tag{5.4}$$

Table 5.10 summarizes the two episodes of agreement analysis for the FE assignment subtask of annotation. We expect that the calculated agreements over the sub-corpora can be generalized to the whole set of annotation with respect to frame evocation and FE assignment.

From Table 5.9 and Table 5.10, the overall agreement on frame evocation for the predicates is much higher than that of the FE assignments. This is expected although this difference is larger than expected. We explain the reasons for the low FE agreement in three aspects:

i) Different annotators' skills on the annotation task results in different standards of annotation which damage the FE assignment task more than the frame evocation process. This happens as the total number of FE assignations is much more than that in terms of frames.

ii) Different annotators' knowledge in frame semantics, and more specifically in FrameNet, initiates different understandings of the annotation task. Again, this more strongly affects the FE assignment task as there are many FEs with different definitions in FrameNet.

each other exactly or partially. For example, the strings "car crash" and "car crash," only partially match with each other. We do not set any distance measure to limit the extent to which string pairs can match.

iii) Dissimilar interpretations of the sentences and clauses by the annotators yield an undesired bias in annotations.

5.4 Experiments with Different Parsing Levels

Using the experimental QA system described in section 3.2.3, the impact of the different levels of shallow frame semantic parsing, defined in section 5.2, on the performance of factoid answer processing is measured. The FSB model identifies the answer candidates according to the FrameNet-based alignment also explained in Chapter 3. It should be noted that the FE matching in all of the experiments in this chapter is based on a specific method in which there is no necessity for all of the FEs in match frames to contain exactly or partially the same arguments. This will be further explained in Chapter 6.

5.4.1 Initial Runs

The first group of QA runs are based on the subset of 143 TREC 2004 factoid questions the results of which are shown in Table 5.11 and Table 5.12. The first table contains the results of the answer processing module with Merged (FSB-ENB-fused) setting and the second table shows the results of the FSB-only setting (see Chapter 3).

The answer merging strategy in the Merged (FSB-ENB-fused) setting is based on the answer scores. The two lists of answers (retrieved by FSB and ENB) are concatenated and sorted according to the answer scores. Finally, the single top ranked answer is reported by the answer processing module (the detailed description in Chapter 8). The SHALMANESER parser in these experiments is trained with the FrameNet 1.2 dataset and the manual correction uses the FrameNet 1.3 dataset.

Table 5.11: First QA runs on 143 TREC 2004 factoid questions - with Merged (FSB-ENB-fused)

QA and parsing level	*mrr*					
	strict			lenient		
	FSB	ENB	Overall	FSB	ENB	Overall
Baseline answer processing (BL)	N/A	**0.266**	0.266	N/A	**0.280**	0.280
BL + SHAL	0.000	0.238	0.238	0.000	0.252	0.252
BL + SHAL-VF	0.084	0.189	0.273	0.098	0.203	0.301
BL + SHAL-AF	0.119	0.168	0.287	0.140	0.182	0.322
BL + SHAL-HL	**0.217**	0.098	**0.315**	**0.245**	0.112	**0.357**

By comparing the results of our QA system with those participating in the TREC 2004 competition, it can be seen that the performance of our system is far lower than that of the best-performing system on the same subset of questions. Table 5.13 shows the results of the top ten TREC 2004

Table 5.12: First QA runs on 143 TREC 2004 factoid questions - with FSB-only

Parsing level	*mrr*	
	strict	lenient
SHAL	0.000	0.000
SHAL-VF	0.105	0.119
SHAL-AF	0.154	0.175
SHAL-HL	**0.308**	**0.336**

Table 5.13: TREC participant runs and our best run on 143 TREC 2004 factoid questions

Run tag	Submitter	*mrr*	
		strict	lenient
lcc1	Language Computer Corporation	**0.867**	**0.902**
NUSCHUA1	National University of Singapore	0.769	0.797
uwbqitekat04	University of Wales, Bangor	0.671	0.699
IBM1	IBM Research	0.427	0.448
irst04higher	ITC-irst	0.420	0.455
mit1	MIT	0.420	0.441
mk2004qar1	Saarland University	0.413	0.434
FDUQA13a	Fudan University (Wu)	0.315	0.413
BL + SHAL-HL	Our best run (not submitted to TREC)	0.315	0.357
KUQA1	Korea University	0.308	0.308
shef04afv	University of Sheffield	0.294	0.322

participants and our best run. The *mrr* values are calculated with our software evaluator system that examines the output answer lists based on the TREC-reported answer patterns.

5.4.2 System Error Analysis

Because of the low performance of our QA system, we perform a rigorous error analysis on the system. One main reason is found to be the low performance of the baseline ENB module which cannot retrieve many (types of) named entities. With respect to the frame semantic-based model as the main model under consideration, however, there are a number of reasons. Table 5.14 summarizes the result of the error analysis when considering the SHAL-HL annotation level in the frame semantic-based model. From this table, it can be seen that the issue of coverage is one of the biggest problems which interfere with the QA task (R_1 and R_2). After the coverage issue, frame redundancy (R_3) is affecting the task most, followed by the different answer frames than the question frames (R_4).

The reason formulated in R_4 is a challenging problem which requires inter-frame relations beyond the set of frame-to-frame relations which exist in the FrameNet dataset. As an example of such a problem, a TREC 2004 question and its answer passage are considered in Example 5.1 and Example 5.2.

Table 5.14: Error analysis on the frame semantic-based model of the experimental QA system

Reason	Questions affected	
	#	%
R_1: Question does not evoke any frame	18	12.587
R_2: Question does not evoke the main predicate frame	12	8.391
R_3: Passages evoke more than one matching frame and the correct answer is not covered by frames evoked in highly-ranked passages	19	13.286
R_4: Different answer frames compared to question frames	13	9.090
R_5: Answer strings do not match the TREC answer patterns	9	6.293
R_6: Passages do not have predicate-argument answer structure	13	9.090
R_7: Different scenarios in questions and answer passages	12	8.391
R_8: Negative (false) answer redundancy	1	0.699
R_9: Other	1	0.699
Total	98	68.531

Example 5.1-

The question "When was the organization started?" (Q5.2 in the TREC 2004 QA track) is submitted to the passage retrieval module and one of the top 10 passages retrieved contains the answer sentence below:

> Founded in 1956 as an offshoot of the National Retired Teachers Association, AARP is the largest dues-paying organization in the country, with about 32 million members age 50 and up.

The main predicate "start" in the question evokes the frame "Process-start" from the FrameNet database. However, the answer sentence with the predicate "found" invokes a different frame "Intentionally-create" which is not connected to the frame "Process-start" via any existing frame-to-frame relation in FrameNet. The different frames in the question and answer passage cannot match; therefore, the FrameNet-based alignment to extract the candidate answers fails on such frame mismatches.

Example 5.2-

The question "What industry is Rohm and Haas in?" (Q12.1 in the TREC 2004 QA track) returns with an answer passage as below:

> Rohm and Haas, with $4 billion in annual sales, makes chemicals found in such products as decorative and industrial paints, semiconductors and shampoos.

The main predicate "industry" in the question evokes the frame "Fields" which is different from the frame "Manufacturing" that is invoked in the answer sentence by the predicate "make". As a result, the frame alignment strategy cannot identify the matching frame to locate the answer "chemicals" by aligning similar FEs.

The other dominant reason for the FSB model to fail in answer identification is formulated in R_6 where the answer span in the passages does not come with a predicate-argument structure that can

be annotated with the FrameNet elements. Example 5.3 and Example 5.4 show instances of such textual constructions.

Example 5.3-

The question "What rank did he reach?" (Q40.5 in the TREC 2004 QA track) with the target topic "Chester Nimitz" is submitted to the system and the passage retrieval module returns the answer passage below:

> In July, a month after the successful Allied invasion of Normandy, he traveled to Hawaii to meet with Admiral Chester W. Nimitz and General Douglas MacArthur about the conduct of World War II, then toured Pacific military bases.

In this example, the answer string "Admiral" is attached to the target topic string as an adjectival modifier. "Admiral" is not covered by any FrameNet frame; therefore, it does not evoke any frame from the FrameNet dataset. The non-frame-evoking adjectival constructions like "Admiral Chester" interfere with any type of frame matching and FE alignment to pinpoint any answer candidates.

Example 5.4-

The question "What kind of ship is the Liberty Bell 7?" (Q57.1 in the TREC 2004 QA track) returns with the passages one of which contains the answer sentence:

> Newport plans to retrieve the recovery vessel first, then go after Liberty Bell 7, the only U.S. manned spacecraft lost after a successful mission.

One of the answers to this question, reported by TREC, is the string "spacecraft" that occurs in the answer sentence above. However, there is no predicate-argument structure that captures the answer span in the sentence and relates the string "spacecraft" to the ship named Liberty Bell 7. Therefore, the FrameNet-based answer processing mechanism with frame and FE alignment fails in returning the answer candidate.

The problems that arise for the different reasons formulated in Table 5.14 can be handled by different approaches. The major problems and their possible solutions include:

- **R_1 and R_2:** are related to the lexical coverage of FrameNet. One approach to overcome the existing lack of coverage could be to utilize the Detour system (Burchardt, Erk, and Frank 2005) to evoke near frames in the questions. However, to maintain the consistency of annotation in questions and passages it is necessary to perform the same procedure on the passage corpus. This was beyond the time frame of this thesis. Instead, we study the impact of lexical coverage in FrameNet on the answer processing performance in Chapter 7.
- **R_3:** is a technical problem that can be tackled by implementing different procedures of answer processing using frame semantic alignment. In Chapter 6, this will be studied.

- **R_4 and R_7:** require a complicated frame-to-frame relational analysis where there are no FrameNet-based inter-frame relations available to connect two different frames in questions and answer passages. There are two issues to be resolved in this respect, which are not solved in this thesis:
  - Formulating a similarity measure that captures semantic relatedness of two separate FrameNet frames, and
  - Implementing a valid procedure that aligns different FEs of the two semantically related frames.
- **R_5:** is again a technical problem that we try to solve partially using heuristics that improve answer candidates. For instance, we remove prepositions at the beginning of the answers. This problem is not completely resolved in our implementations.
- **R_6:** is a situation where frame semantics (and more generally, any predicate argument-based structure) seems to be inefficient. We do not study these situations in this thesis.

Therefore, the input question set TREC 2004 is reduced according to the figures in Table 5.15.

Table 5.15: Filtering the experimental dataset for experiments

Dataset	Total # of items	No answer @10 passages	R_1	R_2	R_4	R_6	R_7	Remaining
tec04 factoid question set	230	87	18	12	13	13	12	75

5.4.3 Final Results

With the final set of 75 TREC 2004 factoid questions, the results of the different QA runs are shown in Table 5.16 and Table 5.17 with respect to the different levels of FrameNet-based annotation.

Table 5.16: QA runs after finalizing the input question set (75 TREC 2004 factoid questions) - with Merged FSB-ENB-fused

QA and parsing level	*mrr*					
	strict			lenient		
	FSB	ENB	Overall	FSB	ENB	Overall
Baseline answer processing (BL)	N/A	**0.400**	0.400	N/A	**0.413**	0.413
BL + SHAL	0.000	0.347	0.347	0.000	0.360	0.360
BL + SHAL-VF	0.160	0.253	0.413	0.187	0.267	0.453
BL + SHAL-AF	0.227	0.213	0.440	0.267	0.227	0.493
BL + SHAL-HL	**0.413**	0.107	**0.520**	**0.467**	0.120	**0.587**

The results on the 75 TREC 2004 factoid questions at the BL+ SHAL-HL level in the Merged (FSB-ENB-fused) setting and at SHAL-HL in the FSB-only setting show that the rank of our

Table 5.17: QA runs after finalizing the input question set (75 TREC 2004 factoid questions) - with FSB-only

Parsing level	*mrr*	
	strict	lenient
SHAL	0.000	0.000
SHAL-VF	0.200	0.227
SHAL-AF	0.293	0.333
SHAL-HL	**0.587**	**0.640**

Table 5.18: TREC participant runs and our best run on the selected 75 TREC 2004 factoid questions

Run tag	Submitter	*mrr*	
		strict	lenient
lcc1	Language Computer Corporation	**0.867**	**0.893**
NUSCHUA1	National University of Singapore	0.827	0.867
uwbqitekat04	University of Wales, Bangor	0.720	0.733
Irst04higher	ITC-irst	0.547	0.587
BL + SHAL-HL	Our best run (not submitted to TREC)	0.520	0.587
mit1	MIT	0.520	0.533
mk2004qar1	Saarland University	0.480	0.480
IBM1	IBM Research	0.453	0.453
FDUQA13a	Fudan University (Wu)	0.413	0.507
KUQA1	Korea University	0.360	0.360
shef04afv	University of Sheffield	0.347	0.387

QA system improves among the top ten TREC 2004 participants. It is also observed that our experimental QA system is now closer to the best-performing TREC 2004 system with respect to the answer processing *mrr* values. The results of the top ten TREC 2004 participants and our best run, on the set of selected 75 factoid questions, can be seen in Table 5.18. The ranking of our system among the top ten TREC 2004 participants, however, is not a perfect indication of its real performance compared to the TREC participants. This is because we have not used the whole set of factoid questions in the TREC QA track for the evaluations.

5.5 Discussion

The results obtained in Table 5.11 and Table 5.16 and also in Table 5.12 and Table 5.17 have the same trends with respect to the changes over the *mrr* values for both frame semantic-based and entity-based models as well as the overall QA performance. However, the results in Table 5.16 and Table 5.17 on 75 questions are generally at higher levels of performance. Since the strict and lenient measures have the same trends as well; therefore, only the strict measures in Table 5.16 and Table 5.17 are discussed in this section.

The first observation from Table 5.16 is that the maximum performance of the frame semantic-based and entity-based models are achieved in an opposite fashion. That is, when the entity-based model performs its best, the frame semantic-based model has its lowest performance and vice versa. When the entity-based model performs its best, the frame semantic-based model is not yet incorporated into the QA system. As the frame semantic-based model is added to the system, it starts to dominate the entity-based model. This dominance gradually increases as the frame semantic-based model gets the chance to work on the texts with richer shallow semantic information at higher levels of parsing. Eventually, the maximum performance of the frame semantic-based model results in the maximum overall performance of the QA system which is not the case when the performance of the entity-based model is maximum. This shows the importance of the frame semantic-based model and how it can assist with elevating the overall QA performance.

Second, in a sparsely annotated dataset, the frame semantic-based model cannot identify any answers as in the BL+SHAL and SHAL levels in Table 5.16 and Table 5.17 respectively. At the same time, the entity-based model is negatively affected and performs poorly compared to the BL level in Table 5.16.

Third, with the higher levels of annotation with the FrameNet frames and FEs, the frame semantic-based model identifies more correct answer candidates. In the presence of frame redundancy, the frame semantic-based model still can improve from 0.000 in BL+SHAL to 0.413 in BL+SHAL-HL (Table 5.16) and from 0.000 in SHAL to 0.587 in SHAL-HL (Table 5.17). This shows how any frame semantic-based answer processing module working on the basis of frame semantic alignment can be dependent on the accuracy of shallow frame semantic parsing.

Fourth, the improvement that is achieved over the *mrr* values for the frame semantic-based model has a slightly different rate according to the two subtasks of frame evocation and FE assignment in Table 5.16. In other words, the progress from 0.000 to 0.227 after manual correction of the FEs is more than that from 0.227 to 0.413 after manual frame evocations and corrections. It should be noted that the manual correction of the frames in BL+SHAL-HL includes a complete human level FE assignment at the same time. Therefore, the improvement from 0.227 to 0.413 implies the effect of the FEs as well. As a result, the task of semantic role labeling is shown to have greater influence on the task of frame semantic-based answer processing compared to the task of semantic class identification. By considering the results in Table 5.17, however, this is not observed since the improvement after frame-oriented corrections is almost the same as that of the FE-oriented corrections. This is again a result of both frame-oriented corrections and complete FE assignments.

Fifth, it is observable that the *mrr* values of the frame semantic-based model rise when verb frames and non-verb frames are manually augmented with their FE assignations. The rise from 0.160 in BL+SHAL-VF to 0.227 in BL+SHAL-AF in Table 5.16 is evidence for the effect of other part-of-speech frames which are fully corrected with respect to their FEs. The same scenario is repeated in Table 5.17 by the difference from 0.200 in SHAL-VF to 0.293 SHAL-AF. While the impact of

non-verb frames has been studied very little in the literature, our results show the importance of them in frame semantic-based answer processing. Since non-verb predicates participate in improving answer processing performance, an advantage can only be achieved by using FrameNet compared to other linguistic resources such as PropBank and VerbNet which only encapsulate verb semantic frames.

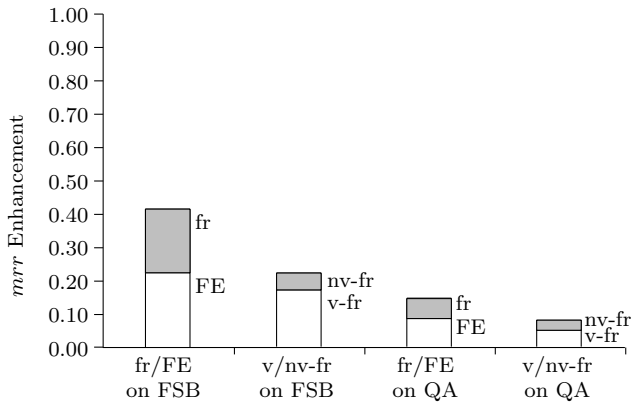

Figure 5.6: The different contribution rates to the *mrr* values in the Merged (FSB-ENB-fused) setting, fr: frame, FE: frame element, v: verb, nv: non-verb

Sixth, the improvement in the *mrr* values of the frame semantic-based model elevates the overall QA performance in Table 5.16 which is promising. This is quantified by the difference between the BL+SHAL level where the overall *mrr* is 0.347 and the BL+SHAL-HL level where this rises to 0.520. Therefore, the positive effect of the usage of the frame semantic-based model in the overall QA performance is realized by the higher levels of semantic class identification and semantic role labeling.

From Table 5.16 there are two more observations with respect to the overall QA performance:

- The pattern of the effect of the individual subtasks of frame evocation and FE assignment and the different part-of-speech frames on the overall performance of the QA system is similar to that of the effect of the same issues on the performance of the frame semantic-based model. That is, FEs are influencing the answer processing *mrr* values more than frames.
- Both verb frames and non-verb frames have an impact on the overall *mrr* measures.

Figure 5.6 and Figure 5.7 visualize the different contribution rates of these issues on both the frame semantic-based model and overall QA performances considering the strict evaluation paradigm in both settings of Merged (FSB-ENB-fused) and FSB-only respectively. The contribution of verb/non-verb frames is measured with respect to the values obtained by BL+SHAL-AF and SHAL-AF.

Finally, there is an important observation when considering the maximum overall *mrr* values

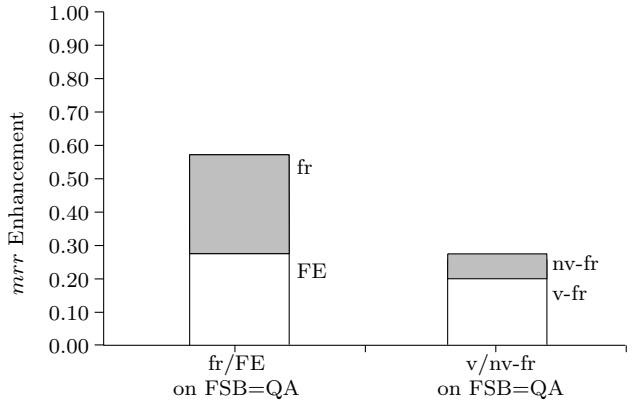

Figure 5.7: The different contribution rates to the *mrr* values in the FSB-only setting, fr: frame, FE: frame element, v: verb, nv: non-verb

of Table 5.16 with those in Table 5.17. The maximum overall *mrr* of the Merged (FSB-ENB-fused) setting reaches to 0.520 while this measure in the FSB-only setting goes up to 0.578. This phenomenon leads us to more carefully study the impact of the two models on each other in order to obtain the maximum benefit from the different capabilities of the models. In Chapter 8, this issue will be investigated.

5.6 Summary

The impact of the different levels of shallow frame semantic parsing on the task of factoid answer processing has been studied in this chapter. To address these research questions, four levels of parsing have been defined with respect to the subtasks of frame evocation and FE assignment in FrameNet-based parsing. In addition, two levels of verb frames and non-verb frames have been considered to observe the impact of the different part-of-speech predicates on the performance of factoid answer processing based on frame semantics. To conduct the experiments, a comprehensively annotated answer passage corpus and an annotated question corpus have been constructed using SHALMANESER and human expert annotators.

The results of our different QA runs with the two frame semantic-based and entity-based models have been discussed and shown that the performances of the frame semantic-based answer processing model and the overall QA system are dependent on the levels of shallow frame semantic parsing as well as the different part-of-speech predicates.

An important observation from the experimental results in this chapter, related to the answer fusion process, has led to a more careful analysis which will be described in Chapter 8.

To see how different methods of frame semantic-based answer processing may result in higher QA performances, in Chapter 6 a set of different methods of frame semantic alignment for answer

extraction and scoring will be studied.

Chapter 6

FrameNet-Based Answer Processing Techniques

Factoid answer processing using the frame semantics encapsulated in FrameNet can be performed in different ways. By exploiting the scenario-based relations in the frames of FrameNet and by focusing on the semantic roles which participate in the events of the frames, it is possible to use different approaches of pinpointing answer candidates to factoid questions. Furthermore, the answer scoring can be carried out in different ways to rank and report the answers. Each answer processing technique using FrameNet may also lead to a different set of answer candidates per given question.

There are advantages over other approaches of answer processing (such as NE-based processes) when using FrameNet-based pieces of evidence for finding exact answer spans:

i) In a comprehensively annotated environment, it is not necessary to filter the FrameNet-based answer candidates according to their NE category. This is because the task of semantic role labeling is supposed to have handled the argument detection and assignation procedure completely and correctly. As a result, the instance values of matching FEs in answer passages cover the right textual window of answer candidates. NE filtering can, however, be performed for identifying exact answers in the presence of complicated textual affixes such as articles and prepositions that may be attached to answer candidate strings.

ii) With different part-of-speech target predicates covered together in FrameNet frames, there is no need for further normalization of predicates such as deverbals, de-adjectival nouns, and de-adverbial nouns.

iii) Non/multiple-NE-categorized answer candidates are also able to be extracted such as the answers to *why* and *how* questions. The REASON answers and MANNER answers may either not contain any NE or include more than one NE of different types. Therefore, the NE-based techniques cannot handle the situation. By using the FrameNet-based alignment techniques it is possible to find such answers as the instance values of the FEs such as "reason" and "manner". These FEs can be found in many FrameNet frames and may be used for aligning the non/multiple-NE-categorized arguments.

iv) Due to the boundaries of the arguments of the predicates in answer passages, the FEs may contain a full representation of the piece of information sought. That is, the complementary information which may come in relative clauses is also originally included in the answer snippets, although it is possible to remove them and return the exact and succinct answers.

In this chapter, a range of techniques to answer candidate identification and scoring are described and their practical outcomes are discussed[1]. The analysis of the experimental results suggests the strongest technique.

6.1 FrameNet-Based Alignment Methods

Identification of the answer candidates to factoid questions using FrameNet can be attained using different levels of linguistic information as well as different techniques at each linguistic level. This chapter concentrates on semantic approaches that utilize frame semantics alignment. This type of alignment can be viewed from two perspectives with respect to the specifications of frame semantics:

i) The scenario-level evidence which is realized by matching the semantic frames in FrameNet, and
ii) The argument or semantic role-level evidence which is performed through matching the FEs corresponding to the FrameNet semantic frames.

Generally, the main goal of frame-level matching is to identify parts of the answer passages which are centred around the same event mentioned in a given question. Therefore, the first level of semantic matching emphasizes the broad picture of the events and can provide a conceptual scenario-based normalization over the texts of questions and answer-bearing passages. This normalization results in handling different types of text paraphrasing that may have interfered with identification of the answer candidates concealed by a surface syntactic mismatch.

The second level of evidence, however, can be used to pinpoint the exact spans of the answer passages that refer to the focus of the question. In other words, it is a process of matching semantic roles that participate in similar events of the question and answer passages. This can also be exploited just for scoring the answer candidates retrieved via other techniques (Shen and Lapata 2007).

Figure 6.1 shows the two levels of evidence for answer candidate identification based on frame and FE matching.

The use of the two levels of FrameNet-based evidence for factoid answer processing can be found in the literature (Fliedner 2004; Hickl et al. 2006; Kaisser 2005; Kaisser, Scheible, and Webber 2006; Kaisser and Webber 2007; Narayanan and Harabagiu 2004*a*; Narayanan and Harabagiu 2004*b*; Shen

[1] The work in this chapter has been partially covered in (Ofoghi, Yearwood, and Ma 2008*a*) and (Ofoghi, Yearwood, and Ma 2009).

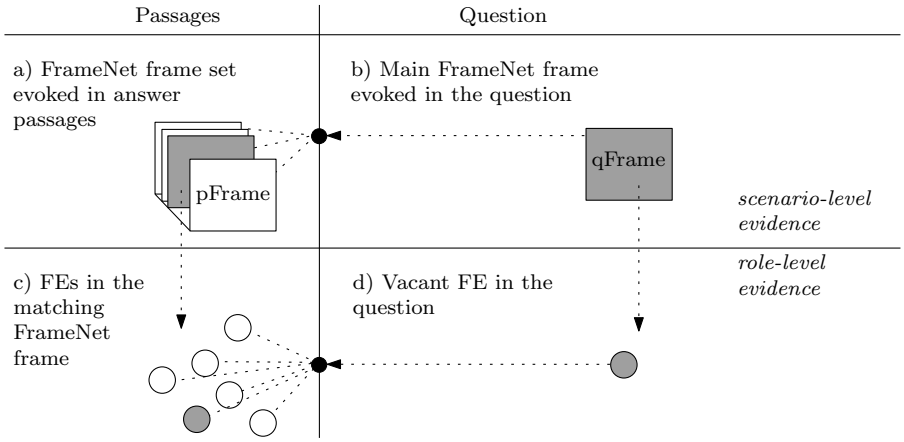

Figure 6.1: Different levels of evidence for answer candidate identification based on FrameNet; a) annotated passage region, b) annotated question region, c) the FEs in the match passage frame, and d) the vacant FE in the question frame

and Lapata 2007). They use different methods for assigning frames and FEs to texts and most of them use frame and FE matching techniques for answer candidate identification. The only technique that identifies answer candidates on the basis of NEs and noun phrases (with no frame and FE matching) can be found in (Shen and Lapata 2007). These have been discussed in detail in Chapter 2.

In the next sections, we introduce a range of semantic alignment techniques where we implement different schemes across both levels of FrameNet-based evidence. We consider the first of these techniques (see section 6.1.1) as the baseline technique since it is the most obvious technique of frame semantic alignment. It is not clear whether the second technique (explained in section 6.1.2) has been used by other researchers. Some similar (but not the same) approaches can be found in the previous works mentioned earlier. The other techniques (discussed in sections 6.1.3 to 6.1.5) are completely different from existing methods and are new to answer processing using FrameNet. par In Figure 6.2, a technical view of the frame semantic alignment method is shown where QFrame and PFrame refer to the frames evoked in the question and answer passage respectively. The frame matching and FE alignment procedure are based on exact frame and FE names in all of our implemented techniques. The FE value matching process between non-vacant FEs, however, can be performed on the basis of exact or partial string matches. FE value matching of non-vacant FEs is only carried out in the first technique (see section 6.1.1) and is conducted using a partial string matching procedure in our experiments.

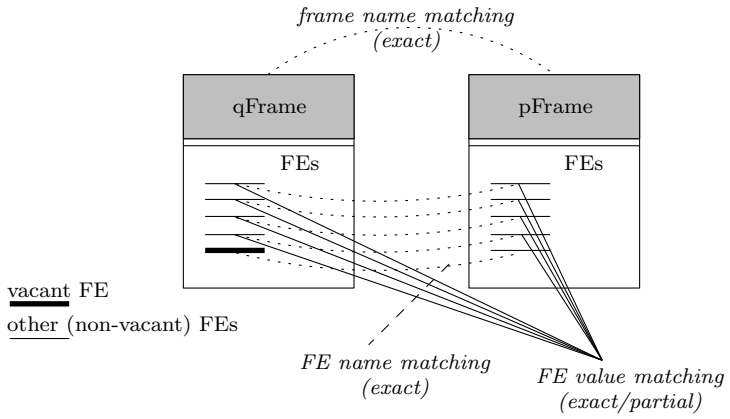

Figure 6.2: FrameNet-based alignment: A technical view

6.1.1 Complete Frame and FE Alignment - No Frame Scoring

In this method (referred to as CFFE), any frames from the top ranked retrieved passages that match with the question main frame(s) are selected. Frame matching is based on matching the name of the frames. In case there is more than a single frame evoked by question predicates, the main frame(s) of the question are selected as those frames which contain a vacant FE. The way of identifying if a given FE is a vacant one is based on the existence of the question stems *how*, *what*, *when*, *where*, *why*, *which*, *who*, *whose*, and *whom* in the instance value of that FE. This could be elaborated by more sophisticated linguistic approaches; however, we do not concentrate on this part and only rely on this straightforward and efficient approach. Our method of identifying main question frames works so effectively that we have not found any misclassification of non-main frames into the main frame category in our experiments.

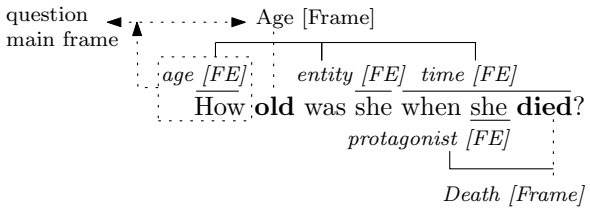

Figure 6.3: Identification of the question main frame

Figure 6.3 shows an example question (Q31.4 in the TREC 2004 QA track) annotated with the two FrameNet frames "Age" and "Death" evoked by the predicates "old" and "die" respectively. Having assigned the FEs of the two frames to the arguments of the predicates, the main question frame is selected to be the frame "Age" as it includes the vacant FE "age" containing the question stem *how*

in its instance value. The frame "Death" is not considered as a main frame since it does not include any vacant FE. All this process is carried out automatically in our experimental QA system.

There is always the possibility of having questions with multiple main frames. If this happens, all of the question main frames are subject to frame and FE alignment in order to identify answer candidates. We have seen instances of this condition in the TREC 2004 experiments.

```
1. foreach(question frame in the question frame set){
2.   if (question frame is a main frame){
3.     if (passage frame set != null){
4.       foreach(passage frame in the passage frame set){
5.         if (passage frame is a matching frame){
6.           if (question frame and passage frame match in all FEs){
7.             foreach(question FE in the question FE set){
8.               if (question FE is a vacant FE){
9.                 foreach(passage FE in the passage FE set of passage frame){
10.                 if (passage FE's name == question FE's name){
11.                   answer candidate = passage FE's instance
                      value…}}}}}}}}}}
```

Figure 6.4: Frame and FE alignment in the CFFE method for answer candidate identification

Once the passage frames are filtered with respect to the question main frame(s), the FE alignment step is executed on each of the individual frames in the filtered set. In this step, each of the FEs for each passage frame is aligned with its corresponding FE in the question main frame. In this method, a frame is referred to as a matching frame if all of the instance values assigned to the FEs in both the passage frame and the question frame are the same except for the vacant FE. In a matching passage frame, the instance value of the FE which corresponds to the vacant question FE is considered as an answer candidate. Figure 6.4 shows the pseudo code of this method which summarizes the different stages of frame and FE alignment.

Line number 6 is where the complete FE matching is performed by comparing the instance values of the pairs of the FEs from the passage and question frames. Since the FE matching process is performed on the basis of the textual strings assigned to the FEs, there are challenges that can interfere with the matching performance. The main problem is related to the assignation of some FEs to parts of texts which partially differ from each other. For instance, the string value "in 1932," partially matches the string "1932". If the FE matching process is conducted to strictly match the strings, a considerable number of partial matches will be ignored which will result in a much lower FE matching performance. Therefore, in our implementation of the CFFE method, a partial string matching procedure is used for FE alignment. The procedure for partial matching is shown in Figure 6.5.

Complete matching of the set of FEs implies a rigorous condition that drastically reduces the number of passage frames. By having very few passage frames (in most cases 0 or a single frame) the chance of matching frame redundancy is small. As a result, it is not necessary to overload the

CFFE method with a frame scoring scheme to select particular matching frames and rank them according to more complicated criteria. Consequently, in our implementation of the CFFE method, there is no frame scoring process for the strictly matching passage frames to question main frames.

```
1. result = true;
2. foreach(question FE in the question FE set){
3.   foreach(passage FE in the passage FE set of passage frame){
4.     if (question FE's name == passage FE's name){
5.       if (question FE is NOT a vacant FE){
6.         if ((question FE == "") XOR (passage FE == ""))
7.           result = false;
8.         prepare question FE's instance value (qiv);
9.         prepare passage FE's instance value (piv);
10         if ((qiv does NOT contain piv) OR (piv does NOT contain qiv))
11           result = false;}}}}
12.return result;
```

Figure 6.5: FE matching between a passage frame and a question main frame

The answer candidates which are identified by this method are scored only based on the scores of the answer-containing passages. The passage scoring function is part of the passage retrieval module explained in Chapter 3 and Chapter 4.

6.1.2 Frame Alignment with Specific FE Matching - No Frame Scoring

To relax the requirement of matching all FEs in the CFFE method, the main condition for the question main frames and passage frames to be matching frames is changed in this method (known as FSFE-NFS). As one of the existing approaches which performs relaxation on semantic role matching, the work by Kawahara, Kaji, and Kurohashi (2002) treats semantic predicate-argument structures of questions and passages as match structures if they share at least one argument.

In our work, we relax the matching requirement so there is no necessity for the FEs in the passage frames and question main frames to match. Instead, a passage frame is called a matching frame with a question main frame only if the two frames share the same name. The frame and FE alignment procedure is shown in Figure 6.6. The only difference between the alignment process of FSFE-NFS and the alignment strategy of the CFFE method, shown in Figure 6.4, is that the frames are not checked for FE matching (as in the line number 6 of Figure 6.4).

The procedures for identifying the vacant question FEs and finding an answer candidate in the FSFE-NFS method are all the same as in CFFE. In this method, also, there is no frame scoring scheme considered. Like the CFFE method, the answer candidates are scored only based on the answer-bearing passages.

```
1. foreach(question frame in the question frame set){
2.   if (question frame is a main frame){
3.     if (passage frame set != null){
4.       foreach(passage frame in the passage frame set){
5.         if (passage frame is a matching frame){
6.             foreach(question FE in the question FE set){
7.               if (question FE is a vacant FE){
8.                 foreach(passage FE in the passage FE set of passage frame){
9.                   if (passage FE's name == question FE's name){
10.                    answer candidate = passage FE's instance
                       value…}}}}}}}}}
```

Figure 6.6: Frame and FE alignment in the FSFE-NFS method for answer candidate identification

6.1.3 Frame Alignment with Specific FE Matching - Frames Scored

With respect to the relaxed frame matching strategy taken in the FSFE-NFS method, there will be a number of matching passage frames with the main question frames in the filtered set of passage frames. Therefore, a comprehensive frame scoring procedure on the basis of any possible pieces of evidence may assist the answer processing module in correct answer retrieval. In this method (FSFE-FS), such a scoring procedure will have at least two benefits to the task:

i) The frame scoring scheme can reduce the negative impact of frame redundancies in the answer passages by differentiating the frames which occur in the same passages via different scores, and
ii) It overcomes the problem of first-occurred-higher-scored[2] frames and possible answer candidates in the answer passages by making the scores less dependent on the initial passage-based scores.

To more comprehensively score the passage frames which have already been assigned the scores of their containing passages, we use two types of evidence:

i) The instance value of the FE which corresponds to the vacant question FE - if there is an instance value (not null and not an empty string) assigned to the FE of the passage frame corresponding to the vacant question FE in the question frame, then we add 1.0 to the initial score (Figure 6.7).
ii) Query term frequency - the score of a frame is added up with the raw term frequency of each query term - formed on the basis of the question - in the frame-bearing sentence (Figure 6.8).

It should be noted that in line 7 of Figure 6.7, the passage frame is just qualified and the score is not changed yet. With this qualification, its score will be changed only once in the containing procedure. This prevents the frames from getting additional scores in case there is more than one vacant FEs (as a result of the existence of multiple main question frames) with the same name.

[2] Higher frame scores are due to the higher scores of containing passages which are scored higher and occur higher in the list of retrieved passages.

```
1. if (the FE set of passage frame != null){
2.   foreach (passage FE in the passage FE set of passage frame){
3.     for (int i=0; i<the number of vacant question FEs; i++){
4.       if (passage FE's name == vacant question FE[i]){
5.         if (the value of passage FE != null){
6.           if (the value of passage FE != ""){
7.             passage frame is qualified for score change
8.             break;}}}}}}
```

Figure 6.7: Frame score changing according to the existence of an instance value in the FE which corresponds to the vacant question FE

The first piece of evidence is used to more specifically identify a passage frame which actually and potentially contains an answer candidate that is included as the right semantic role with respect to the event of the answer passage and the question. The second type of evidence is a surface indication of how close the answer sentence and the question sentence are.

```
1. double resultant frequency (rf) = 0.0;
2. for (int i=0; i<the number of query terms; i++){
3.   rf += term frequency of query terms[i] in passage frame-bearing
     sentence;}
4. the score of passage frame += rf;
```

Figure 6.8: Boosting the score of passage frames according to the raw frequencies of query terms

After scoring the passage frames, the task of finding answer candidates is performed using a similar approach to that of the FSFE-NFS. The answer candidates which are extracted, however, will have different scores and rankings compared to the list of answers that can be retrieved by the FSFE-NFS method.

6.1.4 FE Alignment - No FE Scoring

This method (FE-NFES) is a big step towards making the answer processing strategy shallower in the sense of semantic matching. There is no passage-question frame matching performed prior to the FE matching process in the FE-NFES method. The passage FEs are identified as the matching FEs to the question FEs only if they share the same name regardless of the semantic frames which include these FEs. There is no passage FE scoring scheme in this particular method. As a result, the answer candidates - the matching passage FE instance values - are only assigned the scores of their container passages. Figure 6.9 shows the pseudo code of the shallow FE matching process in this method.

The procedure in Figure 6.9 is very similar to that in Figure 6.4. The only difference in Figure 6.9 is that it does not contain lines number 5 and 6 in Figure 6.4 where the frame matching process

```
1. foreach(question frame in the question frame set){
2.   if (question frame is a main frame){
3.     if (passage frame set != null){
4.       foreach(passage frame in the passage frame set){
5.         foreach(question FE in the question FE set){
6.           if (question FE is a vacant FE){
7.             foreach(passage FE in the passage FE set of passage frame){
8.               if (passage FE's name == question FE's name){
9.                 answer candidate = passage FE's instance value…}}}}}}}}
```

Figure 6.9: FE alignment in the FE-NFES method for answer candidate identification - no frame matching is performed before FE alignment

is conducted. The question main frames and vacant FEs are identified in a similar way to that described in section 6.1.1 for the CFFE method.

The number of matching passage FEs to the question (vacant FEs), according to the FE-NFES method, can be large. This method may cover some of the answer candidates that cannot be identified by the methods which follow a frame matching procedure before any FE alignment.

6.1.5 FE Alignment - FEs Scored

So far, matching passage FEs (to the vacant question FEs) are ordered only on the basis of the score of their container passages. There is a need for a scoring scheme that accounts for the dependency of the FEs beyond the scores of their passages. As the number of matching passage FEs grows in the FE-based alignment techniques, due to the shallow semantic matching process with frame matching, a FE scoring task is considered to be more vital.

In this method (referred to as FE-FES), which is a more elaborated version of the FE-NFES method, the FEs are scored based on two pieces of evidence:

i) Score of their parent frame: the parent frames of the FEs are scored with their passage scores and the accumulated query term frequencies as mentioned in Figure 6.8 in section 6.1.3. The frame scores are taken as being the initial scores for the FEs.

ii) Instance values: +1.0 is added to the FE score if its instance value is not null or an empty string.

The two conditions above can move the FEs in their initial ranked list, based on the passage scores, up and down as the FE scores are now less dependent on the passage scores. As mentioned in section 6.1.3, in theory this is a positive impact as it can elevate the score and the rank of a correct answer candidate that is identified in lower ranked passages. Figure 6.10 demonstrates the procedure of FE scoring in the FE-FES method. Line number 4 assigns the frame scores to the FEs, according to the raw query term frequencies in the frame-evoking sentences. The next lines check for the second condition of FE scoring based on the instance values in the FEs.

```
1. foreach(passage frame in the passage frame set){
2.   score passage frame with raw query term-based frequencies;
3.   foreach(passage FE in the passage FE set of passage frame){
4.     passage FE's score = passage frame's score;
5.     if (passage FE's instance value != null){
6.       if (passage FE's instance value != ""){
7.         passage FE's score += 1;}}}}
```

Figure 6.10: FE scoring procedure in the FE-FES method

6.2 Conceptual Analysis of Alignment Methods

The different frame semantic-based alignment techniques described in section 6.1 can be compared with each other from different perspectives in a 4-dimensional space:

- The chance of finding matching passage elements with respect to the criteria which are enforced in the matching processes in each technique,
- The level of semantics that is taken into consideration for identification of the answer candidates,
- The level of dependence of the techniques on the level of shallow semantic parsing of the questions and passages, and
- The overall performance of the techniques in extracting factoid answer candidates.

With respect to the chance of finding the elements which match with question elements, the CFFE method is expected to have the minimum number of matches. The rigorous frame matching procedure in this method prevents it from finding many matching frames. The relaxed frame matching process in the two consequent methods of FSFE-NFS and FSFE-FS results in a greater number of matching frames possible to be identified with the question main frames. This is the main reason for scoring the frames in the FSFE-FS method to more carefully deal with the frame redundancies.

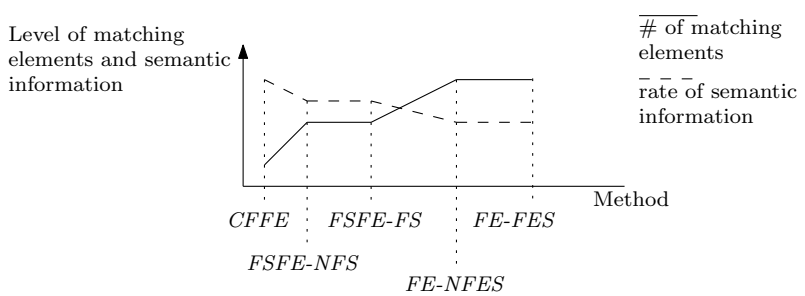

Figure 6.11: The level of matching elements and semantic information in the different FrameNet-based answer processing methods

The number of matching elements in the FE-NFES method grows again due to the shallower matching procedure which is conducted only between the FEs of the passage and question main frames. This number is also the same in the last method - FE-FES. However, the ranking of the final matching passage FEs is different since there is a scoring scheme in FE-FES to overcome the problem of FE redundancies. Figure 6.11 simulates the rate of the matching elements and the semantic information taken into consideration in the five alignment techniques.

From the second point of view - the level of semantic information taken into consideration - there are two general classes with minor differences in the techniques. The first general class includes the more comprehensive frame semantic-based alignment which considers the events (and/or states) as well as the participants in the events. Such an approach towards identifying matching answer candidates to a given question reflects the fact that the candidates are required to participate in the same (or at least similar) scenarios in order for them to be considered as potential answer entities. In other words, the scenario-based relations need to hold and keep the answer candidates as semantically close to each other as possible using the frame semantics encapsulated in FrameNet.

The second class, however, breaks the above-mentioned tie which connects the FEs in a semantic frame together. While this tie ensures that the passage FEs are more semantically related to the vacant question FEs, the absence of any frame matching process before matching the FEs is a step towards making the answer identification task shallower with less semantic information involved. With respect to the characteristics of FrameNet where there are FEs with the same names in different frames (especially non-core FEs like the FE "time"), it is possible for the FEs to match with the same names although they are included in different frames. In such situations, the FE-oriented methods go beyond the boundaries of the FrameNet frames and the semantic information they encapsulate. However, the methods are still bounded to the semantic roles (the FEs of the frames) assigned to the text which keep the methods ahead of the simple information extraction-based methods such as named entity tagging-based ones.

One of the main benefits of this method is realized when a passage frame and a question frame are slightly different but conceptually very similar. For instance, the frames "Receiving" and "Sending" both refer to a similar concept with different perspectives. The FE "recipient" is included in both frames; therefore, it is possible for the "recipient" of a "Sending" question main frame to be matched with the "recipient" FE of the frame "Receiving" in an answer passage.

According to the definitions of the different techniques, the CFFE, FSFE-NFS, and FSFE-FS methods are categorized in the first class of frame and FE-oriented alignment methods and the FE-NFES and FE-FES methods fall into the second class of FE-oriented alignment techniques.

There are minor differences between the members of the first class. The CFFE method differs from the other two methods in the first class to the extent that it implies a more meaning-oriented matching process through aligning all of the FEs of both the passage frame and question frame. The complete matching process of the FEs translates into a perfect scenario-based matching where

all of the participants (except for the vacant question semantic role) are identical. When the frame matching process is relaxed in the two subsequent methods - FSFE-NFS, and FSFE-FS - the scenario-based matching is reduced to capture only the general concept of the events regardless of the event participants.

The level of shallow semantic parsing can also affect the FrameNet-based alignment techniques for factoid answer processing as shown in Chapter 5. However, the different methods are not expected to be equally affected in this regard. By remembering the fact that the shallow frame semantic parsing task consists of the two subtasks of frame labeling and FE assignment, it can be understood that the matching process which is dependent on both of the subtasks is more likely to be affected at different levels of parsing. In the class of FE-oriented methods, even if the frame labeling subtask of parsing goes wrong, there is still some chance of success. This is because some FEs are shared in different frames and even a wrongly evoked frame may contain the desired vacant question FE. On the other hand, if the FE assignment subtask is not performed with a high precision, there is no chance for any of the methods to identify answer spans.

In the first class of frame and FE-oriented alignment, the CFFE method is the one which is most dependent on the levels of parsing. The main reason is that the shallow frame semantic parsing task has to perform perfectly in both answer passages and questions in order for the method to succeed.

In the second class of FE-oriented matching, both of the methods need minimum precision in shallow frame semantic parsing. They can successfully find an answer candidate if two FEs (in the answer passage and the question) are assigned correctly.

The final parameter to compare the FrameNet-based alignment techniques is the overall performance of factoid answer processing. In order to conceptualize this dimension, the best approach is to conduct different QA runs and analyse the experimental results. The results obtained on the basis of a series of QA experiments will shed more light on the different techniques and can be considered in identifying the most effective technique.

CFFE is potentially the most accurate method since it considers the largest amount of semantic information in texts (semantic classes and complete set of semantic roles); however, there are two issues which interfere with its performance:

- Minor mismatches between the FE instances in questions and passages, and
- Entities which are hidden behind a chain of predicative relations in passages.

While the first issue leads us to the design of the FSFE-NFS and FSFE-FS methods, the latter provokes a more comprehensive analysis of the situations where it is possible to semantically resolve predicative relations. This will be studied in the next section.

6.3 Predicate Chains and Complete Frame Semantic Alignment

One of the issues that interferes with CFFE's performance is related to the existence of *predicate chains* in texts of answer-containing passages. By predicate chains we refer to a specific type of lexical chains (Morris and Hirst 1991) which are sequences of semantically related terms in a text. There are three main types of lexical chains with WordNet-based relations: i) *extra-strong* chains, ii) *strong* chains, and iii) *regular* chains.

Extra-strong chains exist between repetitions of the same terms, such as pronouns referring to specific nouns in texts (anaphoric references). Strong chains are constructed between the terms from the same WordNet synsets. The relations in strong chains are synonymy/antonymy, is-a, and inclusion. Regular chains can exist when there is an allowable path between the containing synsets of terms.

With these definitions, predicate chains do not fall in any of the above categories of lexical chains. They differ from other types of lexical chains in the sense that the relations between lexical units (lexemes) in predicate chains are formed on the basis of the concept of predicates[3]. Therefore, predicate chains cannot be handled using existing methods that carry out inference on the basis of WordNet-based lexical chains such as the work by Moldovan et al. (2002). Figure 6.12 represents a predicate chain in an example passage.

a)
In 1974, using beams of electrons and antielectrones, or positrons, Richter discovered [a] particle that came to be called Psi/J. It contained two quarks possessing a previously unknown flavor called charm...

b)

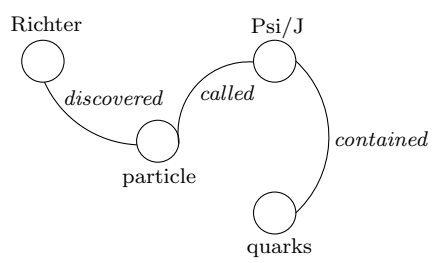

Figure 6.12: A sample predicate chain; a) original text passage, and b) extracted predicate chain between the main entities of the passage

Predicate chains with predicative conceptual relations are not machine understandable or tractable. There are three main reasons for this:

[3] The work in this section has been mainly published in (Ofoghi, Yearwood, and Ghosh 2007).

- Relations in predicate chains carry too much information which is not encoded in machines,
- The number of these relations, based on the predicates and their different senses, is too great which makes the task of understanding the relations computationally ineffective, and
- These relations are not mappable to formally represented inference elements; therefore, no logical or plausible reasoning is possible on them.

Therefore, it is usually not possible to infer new relations between lexical items in texts using predicate chains. Figure 6.13 shows possible indirect relations between the entities of the text in Figure 6.12 which cannot be inferred by existing relations.

As a result of this problem, in the CFFE method of frame and FE alignment for answer processing, it is not possible to extract the correct answer to the question "Who discovered quarks?". This is because in the complete FE alignment procedure of CFFE, the instance values of the FE "phenomenon" in the question and answer passage do not match. Therefore, the correct answer "Richter" cannot be extracted from the answer passage shown in Figure 6.12a. However, this would be rectified if it was possible to infer new relations between the entities in the text of answer passage as shown in Figure 6.13.

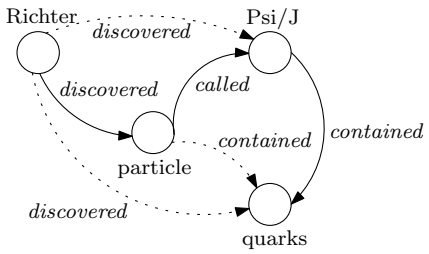

Figure 6.13: New dashed relations are not inferable in predicate chains with predicative relations

Our solution for the problem of non machine tractable predicative relations in predicate chains is based on an ontological extension to FrameNet which generalizes these conceptual relations to a limited set of ontological relations which can be easily mapped to inference elements of a reasoning system. Consequently, using a plausible reasoning system, it is possible to infer new relations between the lexical items in a given text.

6.3.1 Ontologically Extended FrameNet

In FrameNet, there are relations between FEs of two different frames which are realized as the consequences of frame-to-frame relations. Therefore, existing FE-to-FE relations in FrameNet are meaningful since meta-relationships over frames exist. For instance, the FE "perceiver-agentive" in the frame "Perception-active" is a child of the "perceiver" FE in the parent frame "Perception" since the scenario of "Perception-active" is in fact a more specific type of "Perception". These FE-to-FE

relations across FrameNet frames, initiated by FrameNet frame-to-frame relations, have been used in QA in (Scheffczyk, Baker, and Narayanan 2006). The main advantage is that the answering system goes beyond the limitations of specific frame events via the relations using a reasoning module. However, it still cannot be applied where there is no frame-to-frame connection between container frames.

On the other hand, there are three types of frame-internal relations between FrameNet FEs (Lonneker-Rodman 2007):

- **CoreSet:** or Coreness Set is a relation between two or more FEs in a frame such that a sentence with a subset of these FEs is complete. For instance, the FEs "source", "path", "goal", and "direction" are grouped as a CoreSet in the frame "Self-motion". The logical relation "OR" connects the FEs in a CoreSet relation.
- **Requires:** shows the necessity of co-occurrence of two FEs in a frame. This relation indicates a logical "IMPLIES" relation. For example, the FE "entity_1" in the frame "Similarity" *requires* the FE "entity_2". Any sentence with the FE "entity_1" has to have the FE "entity_2" in order to be grammatical. For instance, the sentence "Jack's hair color is similar" suffers from not including the second entity to which "Jack's hair color" resembles. The sentence is grammatical if it is changed to "Jack's hair color is similar to Maria's hair color", for example. In this example, "Jack's hair color" and "Maria's hair color" play the roles of "entity_1" and "entity_2" in the frame "Similarity" respectively.
- **Excludes:** prevents two FEs to occur at the same time in an event. The logical relation "XOR" connects two FEs with an *excludes* relation. The relation between the FE "agent" and the FE "cause" in the frame "Killing" is an *excludes* relation. This means that any killing situation can be realized either by a killer or by a cause.

These existing within-frame relations in FrameNet are limited to the boundaries of the scenarios covered by frames with no (or very narrow) possibility of being used across (non-related) frames. They are not related to any status of the events either.

To overcome such shortcomings, we introduce an ontological extension of frames in FrameNet with respect to the FEs of single frames. The main characteristics of this ontological extension are:

- The relations hold between the FEs in a frame and can be activated across frames using inference engines even where there is no FrameNet frame-to-frame relation between the FE-bearing frames,
- At the main time slices of a complicated scenario, certain ontological relations between the FEs in a frame are valid,
- The ontological relations between the FEs in a frame denote conceptual relationships between participating roles rather than logical connections.

6.3.1.1 Ontological Relation Set

There are some aspects which should be noted when establishing the ontological relation set for predicate chain resolution. First, the relation set has to be a finite set of meaningful relations understandable for other natural language processing communities and knowledge representation and discovery systems. Second, the relations need to be well-defined and machine and human readable. Third, the relations can be *intuitively* correct from a human's point of view or they may be symbolic relations to capture *synthesized* concepts. For instance, the leg of a table is intuitively a part of the table whereas a book on the table at a certain time is a part of that table at that time and this is a synthesized relation.

We have studied the ontological relation set formalized by Bittner, Donnelly, and Smith (2004) containing the foundational relations. These relations are dependent on the entity types and can hold between different types of entities. The entity types are:

- **Individuals:** such as Jules, my car, and her book,
- **Universals:** such as human being, cars, and stars, and
- **Collections:** such as my friends, her previous cars, stars in the Milky Way galaxy.

In addition to the relation set in (Bittner, Donnelly, and Smith 2004), we have inserted the relation *equal-to* which reflects the equality of individual entities. Table 6.1 shows all of the relations where Object1 and Object2 are the signatures (or nodes) of entity type-dependent ontological relations. For example, the relation *individual-part-of* can only hold between individuals, while the relation *member-of* can only be realized when considering an individual and a collection of individuals as signatures. There are some important aspects about the relations:

i) All of these relations (except for the relation *equal-to*) are one-way directed relations,
ii) They are specific to FrameNet frames,
iii) It is possible to have inference over first level relation instances and infer new relation instances (relations at other levels) in a single frame (not covered in our first version of extracted relations),
iv) One alternative approach to find these relations could have been to use the existing semantic types (STs) in FrameNet and their mapping (Scheffczyk, Pease, and Ellsworth 2006) to the SUMO (Niles and Pease 2001) nodes; however, as there are very few STs defined over the FEs in FrameNet, it was not practical to make use of this property of FrameNet. The SUMO relations over STs, also, can not be adapted for the different time slices of complex events.

We chose to start with this set because: i) the time-dependent relation instances in this set make the ontology instances tuned with the exact time frame of events, and ii) the good generalization characteristic of this set translates into a more effective inference over texts using automated inference engines.

The ontological extension can be formalized with respect to the different states of an event. *Stative* frames in FrameNet are regarded as single-status frames while *causative* and *inchoative* frames, which

Table 6.1: Ontological relation set used to extend FN1.2

Object1	Object2	Ontological relation	Example
Individual	Individual	Individual-part-of	You-Your left hand
Individual	Universal	Instance-of	You-Human being
Individual	Collection	Member-of	You-University people
Universal	Universal	Taxonomic inclusion (is-a)	Tiger-Mammal
Universal	Universal	Partonomic inclusion of universals	Animals-Mammals
Collection	Universal	Extension-of	Australian tigers-Tigers
Collection	Collection	Partonomic inclusion of collections	Body parts-Hand parts
Collection	Individual	Partition-of	Your body elements-You
Individual	Individual	Equal-to	You-Your name

are mainly concerned with ongoing events, are treated as multiple-status frames. For the former, the relations hold with no change over time. Considering the latter, however, we emphasize three steps in the events: the *beginning*, *middle*, and *end* of the scenarios. For instance, consider a brief scenario of the frame "Sending" where a "sender" sends a "theme" to a "recipient" in a "container". There are all three statuses in this scenario each of which is related to a particular set of ontological relations between the participant roles. At the beginning, the "theme" is with the "sender", in the middle of the scenario the "container" embraces the "theme", and at the end of the event, the "recipient" owns the "theme". With such a perception, the ontological relations in this scenario are: i) beginning: "theme" *individual-part-of* "sender", ii) middle: "theme" *individual-part-of* "container", and iii) end: "theme" *individual-part-of* "recipient".

Table 6.2: Instances of ontological relations over the FEs in FN1.2

Frame	FE-1	Ontological relation	FE-2
Abounding-with	Theme	Individual-part-of	Location
Buildings	Building	Equal-to	Name
Buildings	Building	Instance-of	Type
Cause-expansion	Item	Member-of	Group
Wearing	Clothing	Individual-part-of	Wearer

For the time being, we formulate the relations at the end of the scenarios for all frames in FrameNet 1.2 in the case of multi-status frames. Table 6.2 shows some instances of the first version of our extension on FrameNet 1.2. There are both types of intuitive and synthesized relations in this table. The entry with the "building" and "type" FEs in the "Buildings" frame shows the intuitive *instance-of* relation while the other entry with "clothing" and "wearer" in the frame of "Wearing" denotes a synthesized *individual-part-of* relation.

The first version of the relation instance set for the FrameNet 1.2 dataset (containing 609 semantic frames) is complete. Table 6.3 shows some statistics about the relations and their frequency of occurrence in this version.

Table 6.3: Statistical information of the relation instances extracted for FN1.2

Ontological relation	Times occurred
Individual-part-of	491
Instance-of	64
Member-of	26
Taxonomic inclusion (is-a), Partonomic inclusion of universals, Extension-of, Partonomic inclusion of collections, and Partition-of	0
Equal-to	87

6.3.1.2 Representation

The axiomatic formalization explained by Bittner, Donnelly, and Smith (2004) is based on a sophisticated theory which can be exploited for the ontological extension of FrameNet, especially with respect to its characteristic of time-dependency which suits the single and multi-status events in the scenarios of FrameNet.

Table 6.4: Part of the ontological extension on the frame "Accoutrements" between the FEs "accoutrement" and "wearer" at the end of the scenario

Ontological relation in XML
<ors>
<or id="158_1" type="individual-part-of" event_status="end">
<signatures>
<s1>accoutrement</s1>
<s2>wearer</s2>
</signatures>
</or>
...
</ors>

At this stage, we do not use this formalization. However, we organize the outcome of the exploration over FrameNet frames in a way which is easy to be added to the FrameNet XML database. Table 6.4 shows an instance of the ontological relation set which has been extracted in the frame "Accoutrements".

6.3.2 Predicate Chain Representation using Extended FrameNet

Focusing on the predicate chain analysis to find semantic connections between the informative pieces of texts, our ontological extension of FrameNet offers a methodology of recognizing relations between NEs and/or information pieces while processing texts on the fly. In the example of Figure 6.12a, there are a few entities: "1974", "beams", "electrons", "antielectrons", "positrons", "Richter", "particle", "Psi/J", "quarks", "flavor", and "charm". The identification of such entities is a task of information extraction systems, while the recognition of their connectivity and semantic relationship is the main

problem tackled in predicate chain analysis. These relations, which are mostly realized by the verb phrases between the entities, lead us to the use of our extended FrameNet as shown in Figure 6.14.

The connectivity between entities of texts is viewed as semantic relations between the participating roles (FE instances) in the scenarios of frames evoked by target predicates. Those FEs of each frame which share the same instance values make the connectivity between the scenarios. In Figure 6.14a, "particle" is the instance value for the FE "phenomenon" of the frame "Achieving-first" and at the same time it is the instance value for the FE "entity" of the frame "Being-named" in Figure 6.14b. This shared instance value initiates the connectivity between the scenarios. Figure 6.14b illustrates all pieces of connectivity between entities in the example text from a FrameNet perspective.

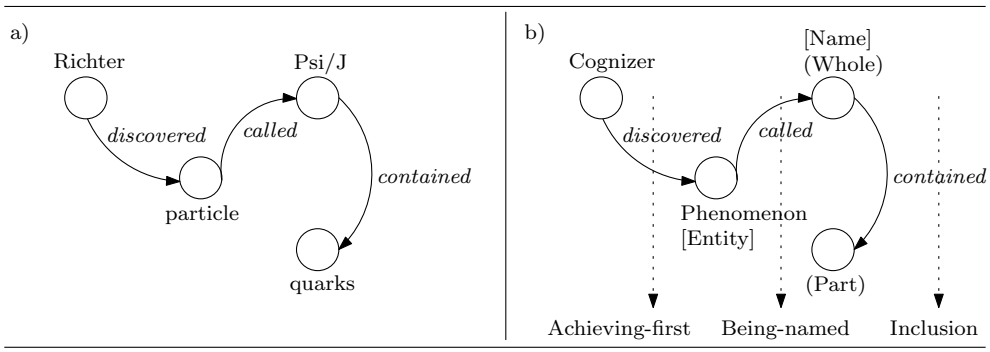

Figure 6.14: Entities and their relations; a) original predicate chain, and b) FrameNet-based mapping of the entities in the original predicate chain

The connectivity shown in Figure 6.14b is meaningful for humans. However, it cannot be exploited by machines to perform automated reasoning and extract new information. It is necessary for the links to be understandable by machines. Our proposed ontological extension on FrameNet offers a sophisticated method of understanding such connections by machines. Figure 6.15 demonstrates the ontological view of the example predicate chain in Figure 6.14a.

The mappings process from Figure 6.14a to Figure 6.15 is based on the ontological relations extracted between FrameNet FEs. The mapping procedure of the relations starts with entry lookup in the list of ontological relations between the FEs in the frame which is evoked by each target predicate. For instance, the target predicate "called" evokes the frame "Being-named". The two FEs "entity" and "name" are related to each other in the list of ontological relations that we have extracted in FrameNet 1.2 via the relation *equal-to*. Therefore, the predicative relation "called" in Figure 6.14 is replaced by the ontological relation *equal-to* in Figure 6.15. In this procedure, the first link ("discovered") is not mapped to any ontological relation since the predicate "discover.v" does not have any relational meaning according to the relation set that we have extracted in FrameNet 1.2.

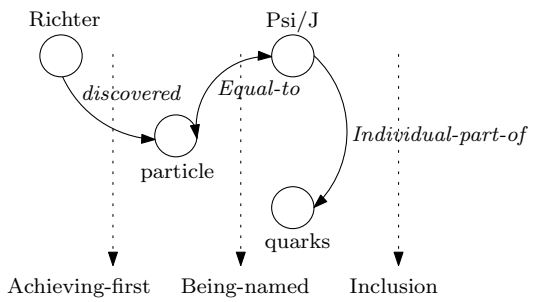

Figure 6.15: Entities and their relations in a predicate chain - Ontological view

6.3.3 Reasoning on Predicate Chains for Answer Processing

In the predicate chain of Figure 6.14a, there are only three directly represented statements corresponding to three questions that can be answered using the CFFE answer processing method:

- Statement-1: "Richter discovered particle" → Question-1: Who discovered particle?
- Statement-2: "Particle called Psi/J" → Question-2: What is particle called?
- Statement-3: "Psi/J contained quarks" → Question-3: What did Psi/J contain?

With the ontological view of predicates it is possible for an automated system to infer more pieces of information. This is brought about because a limited number of ontological relations are mappable to formally represented inference elements. From the ontological view in Figure 6.15, for example, it is possible to infer three new statements so that the CFFE answer processing method can answer three more questions (Figure 6.16):

- Statement-4: "Richter discovered Psi/J" → Question-4: Who discovered Psi/J?
- Statement-5: "Richter discovered quarks" → Question-5: Who discovered quarks?
- Statement-6: "particle contained quarks" → Question-6: What did particle contain?

In order to extract new relations shown in Figure 6.16a, there is a need for a reasoning procedure to interpret existing relations and infer new relations. We propose the use of plausible reasoning (Collins and Michalski 1989) to be applied on the ontological views of texts. In this type of reasoning, plausible inferences are performed over existing knowledge to extract new and reasonable pieces of knowledge. These inferences have been designed based on humans' every day reasoning. In this sense, plausible reasoning is different from formal logic and other types of non-classical logics such as fuzzy logic, multiple-valued logic, Dempster-Shafer logic, intuitionist logic, variable-precision logic, probabilistic logic, belief networks, and default logic (Collins and Michalski 1989).

Statements in plausible reasoning include three main elements: i) *descriptor*, ii) *argument*, and iii) *referent*. For example, the statement "The number of galaxies in the universe is about 125 billion." is shown like "number(galaxy) = ∼125 billion", where "number" is the descriptor, "galaxy" is the

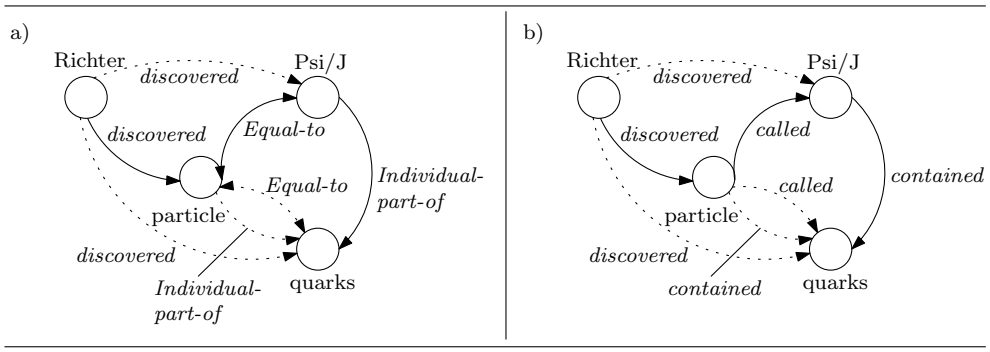

Figure 6.16: New extractable relations on a predicate chain; a) ontological view, and b) predicative view

argument, and "∼125 billion" is the referent. There are also dependency-based logical expressions in the theory of plausible reasoning. Dependency-based expressions formulate relations between different statements. For instance, the expression "The number of stars in galaxies depends on the size of the galaxies." is shown like "number(galaxy_stars)↔size(galaxy)".

Basic inferences in the theory of plausible reasoning are based on the arguments and referents. These include *generalization (GEN)*, *specialization (SPEC)*, *similarity (SIM)*, and *dissimilarity (DIS)* on both arguments and referents[4]. In addition, there are two dependency-based transforms: i) *derivation from dependency (DFDEP)*, and ii) *transitivity inference (TRANS)*. In the DFDEP transformation, the value of each side of the dependency can be inferred from the value of the other side.

Table 6.5: Mapping of ontological relations to basic plausible reasoning inferences

Ontological relation	Plausible transformation
Individual-part-of	GEN/SPEC
Instance-of	GEN/SPEC
Member-of	GEN/SPEC
Taxonomic inclusion (is-a), Partonomic inclusion of universals, Extension-of, Partonomic inclusion of collections, and Partition-of	GEN/SPEC
Equal-to	SIM

Each statement in this theory, and each transformation, has a certainty value. There are a number of different parameters that reflect the certainty of the inference elements. Details of the theory, the full explanations of the transformations, and the list of different certainty parameters in plausible reasoning can be found in (Collins and Michalski 1989).

[4]Inferences on arguments formally start with A and inferences on referents start with R. For instance, RGEN is a GEN inference on referents.

To carry out plausible reasoning on the ontological views of texts for extracting new informative relations, it is necessary that there is a mapping from the ontological relations to the inference transformations. With such a mapping, the inference engine can identify the relationship between arguments or referents and proceed in linking different pieces of information and infer new implicitly available information from texts. Table 6.5 shows this mapping in our proposed solution which is applied for both arguments and referents. In all of the inferences, the TRANS transformation is viable.

In answer processing using the CFFE method, the activation of the plausible reasoning engine is due to an incomplete frame matching where the frame names in a question and an answer passage are the same but the FE instances do not match. Once the reasoning engine is activated on the ontological view of the answer passage, it can proceed in a *forward chaining* or *backward chaining* procedure. Depending on which chaining is implemented, the plausible inferences may be selected by the inference engine to consult appropriate argument or referent-based transformations with respect to the ontological relations. Figure 6.17 shows an example question and two inference procedures which can lead to correct answer identification using the CFFE method.

a) Who discovered quarks?	b) In 1974, using beams of electrons and antielectrons, or positrons, Richter discovered particle that came to be called Psi/J. It contained two quarks possessing a previously unknown flavor called charm...
c) Inf-1: Individual-part-of(Psi/J)=quarks→ Psi/J=quarks [C_1] Inf-2: Equal-to(Psi/J)=particle [C_2] Inf-3: (Inf-2 *and* Inf-1)→quarks=particle [$C_1 \times C_2$]	d) Inf-1: Equal-to(particle)=Psi/J→particle=Psi/J [C_1] Inf-2: Individual-part-of(Psi/J)=quarks→ Psi/J=quarks [C_2] Inf-3: (Inf-2 *and* Inf-1)→particle=quarks [$C_1 \times C_2$]
e) Equal-to(Psi/J)=particle [C_2] Psi/J *AGEN* quarks [C_1] Equal-to(quarks)=particle $f(C_2, C_1) = C_2 \times C_1$	f) Equal-to(particle)=Psi/J [C_1] Psi/J *RGEN* quarks [C_2] Equal-to(particle)=quarks $f(C_1, C_2) = C_1 \times C_2$

Figure 6.17: Inference procedure for resolving predicate chains; C_1 and C_2 are certainty values of the inferences; a) question, b) answer passage, c) backward chaining, d) forward chaining, e) plausible notation of argument-based backward chaining, and f) plausible notation of referent-based forward chaining

The process in Figure 6.17 starts with annotating both the question and answer passage with FrameNet frames and FEs. The CFFE method cannot identify the answer candidate "Richter" since the instance value of the FE "phenomenon" in the question is "quarks" which does not match with the instance value of the same FE of the same frame in the answer passage ("particle"). Therefore, the inference engine is activated to find any possible relation between "quarks" and "particle". Using both forward chaining and backward chaining procedures, it is possible for the inference engine to

extract the new fact that "Richter discovered quarks." with the certainty value equal to C_1C_2. The backward chaining procedure applies the inferences on the arguments while in the forward chaining procedure the inferences are on the referents.

The use of ontological relations that we have extracted for FrameNet in predicate chain resolution is not implemented in the experimental study of this thesis for two reasons:

i) There are not many TREC questions that can be handled using this method while its implementation is complicated and expensive for the QA system,

ii) Predicate chain resolution can only improve CFFE's performance when there are FE mismatches. It cannot be applied to the other FrameNet-based answer processing methods (FSFE-NFS, FSFE-FS, FE-NFES, and FE-FES). In addition, by using the FSFE-NFS and FSFE-FS methods, it is possible to overcome the problem of getting answers that involve predicate chains. These methods can retrieve answers in such situations in a non-semantic way being unaware of the relations and entities behind them. For example, in the case of the example "Who discovered quarks?", after finding the matching passage frame to the question frame, retrieving the instance value of the FE "cognizer" will suffice, although the instance values of the FE "phenomenon" in the two frames do not match. However, the shallow methods in FSFE-NFS and FSFE-FS cannot fully overcome the problem of predicate chains and it is still required that a more comprehensive analysis (such as using our FrameNet-based ontological relations) be performed to pinpoint entities related to each other and semantically resolve FE mismatches.

6.4 Experimental Results

We use the experimental QA system described in Chapter 3 to observe the effectiveness of each answer processing technique mentioned in section 6.1 in the frame semantic-based model of answer processing. The FSB-only setting of answer processing is used because the emphasis in this chapter is on the frame semantic-based techniques. In the CFFE method, there is no semantic resolution of the predicate chains implemented; instead, this method works only on the basis of the definitions in section 6.1.1. The baseline answer processing method (BL), however, uses the ENB-only setting. Answer redundancy is taken into consideration in the lists of answers extracted. The strategy of boosting the rank of the redundant answers is based on the answer scores. The answer scores are modified based on the frequency of occurrence of each single answer in the list of answers. The details of this procedure will be explained in the score-based method in Chapter 8. The results are reported on the basis of the 75 TREC 2004 factoid questions and the 176 TREC 2006 factoid questions selected after a filtering process on the two datasets (see Chapter 3 for more details). There is no manual annotation in the TREC 2006 track; therefore, the entries in the rows of SHAL-VF, SHAL-AF, and SHAL-HL are all N/A for this dataset.

The SHALMANESER parser, for this set of experiments, is trained with the FrameNet 1.2 dataset

whereas the manual annotation task to correct the automated outputs of SHALMANESER is performed using the frameset of the FrameNet 1.3 dataset. Tables 6.6 to 6.10 summarize the results of the QA runs with the five different FrameNet-based alignment techniques in the FSB answer processing model.

Table 6.6: QA runs on 75 TREC 2004 factoid questions using the baseline system (the ENB-only setting) and the CFFE method of answer processing in the FSB-only setting

QA and parsing level	*mrr*			
	trec04		trec06	
	strict	lenient	strict	lenient
Baseline answer processing (BL)	**0.400**	**0.413**	**0.097**	**0.142**
SHAL	0.000	0.000	0.006	0.011
SHAL-VF	0.093	0.107	N/A	N/A
SHAL-AF	0.187	0.213	N/A	N/A
SHAL-HL	0.293	0.347	N/A	N/A

Table 6.7: QA runs on 75 TREC 2004 factoid questions using the baseline system (the ENB-only setting) and the FSFE-NFS method of answer processing in the FSB-only setting

QA and parsing level	*mrr*			
	trec04		trec06	
	strict	lenient	strict	lenient
Baseline answer processing (BL)	0.400	0.413	**0.097**	**0.142**
SHAL	0.000	0.000	0.011	0.017
SHAL-VF	0.200	0.227	N/A	N/A
SHAL-AF	0.293	0.333	N/A	N/A
SHAL-HL	**0.587**	**0.640**	N/A	N/A

Table 6.8: QA runs on 75 TREC 2004 factoid questions using the baseline system (the ENB-only setting) and the FSFE-FS method of answer processing in the FSB-only setting

QA and parsing level	*mrr*			
	trec04		trec06	
	strict	lenient	strict	lenient
Baseline answer processing (BL)	0.400	0.413	**0.097**	**0.142**
SHAL	0.000	0.000	0.011	0.017
SHAL-VF	0.227	0.253	N/A	N/A
SHAL-AF	0.320	0.360	N/A	N/A
SHAL-HL	**0.627**	**0.680**	N/A	N/A

Table 6.9: QA runs on 75 TREC 2004 factoid questions using the baseline system (the ENB-only setting) and the FE-NFES method of answer processing in the FSB-only setting

QA and parsing level	*mrr*			
	trec04		trec06	
	strict	lenient	strict	lenient
Baseline answer processing (BL)	**0.400**	0.413	**0.097**	**0.142**
SHAL	0.000	0.000	0.011	0.017
SHAL-VF	0.173	0.200	N/A	N/A
SHAL-AF	0.240	0.280	N/A	N/A
SHAL-HL	**0.400**	**0.453**	N/A	N/A

Table 6.10: QA runs on 75 TREC 2004 factoid questions using the baseline system (the ENB-only setting) and the FE-FES method of answer processing in the FSB-only setting

QA and parsing level	*mrr*			
	trec04		trec06	
	strict	lenient	strict	lenient
Baseline answer processing (BL)	0.400	0.413	**0.097**	**0.142**
SHAL	0.000	0.000	0.011	0.017
SHAL-VF	0.160	0.187	N/A	N/A
SHAL-AF	0.240	0.280	N/A	N/A
SHAL-HL	**0.413**	**0.453**	N/A	N/A

6.5 Discussion

From the experimental results in section 6.4, the following observations can be made:

- The CFFE answer processing method does not perform well in answer candidate identification because of different types of textual string mismatches and predicate chains. Therefore, the rigorous conditions of FE matching which are conducted after the frame matching process interfere with a high precision in finding answer spans to factoid questions. The lowest result of the QA runs with the different frame semantic-based modules, as a result, corresponds to the CFFE method. The usage of this method, in conjunction with the baseline (BL) method not only cannot elevate the overall QA performance, but also reduces the performance from 0.400 to 0.293 in Table 6.6.
- By relaxing the CFFE method in its FE matching process, the FSFE-NFS method performs better. Once again, this shows that the FE matching procedure, in the presence of many text-related challenges, can be more effectively conduced when only focusing on the vacant FEs rather than all of the FEs of the matching frames. The usage of the FSFE-NFS method in conjunction with the BL method raises the effectiveness of factoid answer candidate identification (from 0.400 to 0.587 in Table 6.7).

- Frame redundancy is another minor barrier in returning correct answer candidates by considering the difference between the performance of the FSFE-NFS and FSFE-FS methods. In the latter, the frame scoring technique elevates the *mrr* values and results in the best overall QA performance among all of the FrameNet-based answer processing techniques (the *mrr* value of 0.627 in Table 6.8). Frame redundancy can occur in an exhaustively annotated environment where all (or most) possible frame evocations and FE assignations are performed. Consequently, the effect of frame redundancy can be more practically observed as the level of annotation improves.
- The FE-based methods, which do not consider the matching frames, perform relatively lower than the FSFE-NFS and FSFE-FS methods. Therefore, the scenario-based relations that are covered in the FrameNet frames are shown to develop an ideal semantic normalization over the texts of the questions and their specifically related passages. Such normalization plays the role of a beneficial meta-relation for FE alignment the lack of which results in a drop of the answer processing effectiveness. By looking at the results, the maximum performance of the FE-NFES method is equal to the BL performance and in the FE-FES method, this value is slightly higher than the BL performance (0.413 in Table 6.10). Both FE-NFES and FE-FES maximum performances are below the FSFE-NFS and FSFE-FS maximum *mrr* values.
- All of the above-mentioned remarks are more pronounced in a comprehensively annotated textual environment where the frame evocations and FE assignations are achieved with high precision values. As the level of annotation decreases, the different FrameNet-based answer extraction and scoring methods do not show much differences. This is more clearly deduced from the QA runs on the TREC 2006 factoid question set in the Table 6.6 to Table 6.10 where there is a sparse annotated corpus available for the answer processing task.

In our experiments, it is shown that a frame matching process prior to the FE alignment task is crucial and can significantly affect the answer processing performance. However, in the presence of different problems which interfere with the performance of the complete FE alignment procedure of CFFE, a relaxed procedure at this stage is preferred. In addition, with many frames evoked in an exhaustively annotated corpus, a frame scoring strategy is shown beneficial in pinpointing the answer spans and ranking the answer candidates in a way which yields more correct answers reported as the first-ranked answer. In our experiments, therefore, the FSFE-FS method is selected as the best-performing FrameNet-based factoid answer processing method.

The FSFE-FS method has shown higher performance than other frame semantic-based approaches in the literature (not pure frame alignment techniques) such as that in (Shen and Lapata 2007) where the authors carry out a similar question filtering task to our filtering process explained in section 5.4.2.

With the existing challenges in using FrameNet for answer processing (such as those explained in section 5.4.2), it is useful to combine the frame semantic-based answer processing method with

other existing methods of extracting and scoring factoid answer candidates. However, a precise combination of the methods requires a comprehensive study and understanding of the situations where each method performs best. Since there is no such information available so far, we carry out an experiment to see how our best frame semantic-based method may impact the best-performing TREC 2004 system. An *artificially* combined processing task is considered where we combine the results of the best frame semantic-based method with those of the best-performing TREC system manually. The combination process is performed in such a way that the second method is activated only if the first method fails in retrieving a correct answer. The results of the two possible combined settings as well as the best-performing TREC system in the TREC 2004 track are shown in Table 6.11. These results show that the combined methods significantly improve the answer processing *mrr* of the best-performing TREC system (LCC's QA system).

Table 6.11: QA runs on 75 TREC 2004 factoid questions - combined settings are constructed by manual judgments of the two answer processing models - p-values after paired t-tests are calculated with respect to the Best-TREC system - values with † are statistically significant ($p < 0.05$)

Answer processing module	*mrr*	
	strict	lenient
Best-TREC	0.867	0.894
Best-FSB	0.627	0.680
Combined (Best-TREC-first)	**0.947** *p=0.007*†	**0.960** *p=0.012*†
Combined (Best-FSB-first)	0.920 *p=0.022*†	**0.960** *p=0.012*†

6.6 Summary

A set of different FrameNet-based techniques for factoid answer processing have been introduced and discussed in this chapter. The techniques benefit from a range of semantic information for pinpointing answer spans in answer passages. The scenario-based information encapsulated in frames and the participant roles in the events are the main pieces of semantic information that can focus the attention of the answer processing module on the exact segments of answer passages where it is more likely for answer candidates to have been positioned.

According to our experiments, the exploitation of scenario-based information (frame scenarios) in conjunction with semantic roles (the FEs of the frames) results in improved performance in identifying correct factoid answers. To maximize this improvement, it is useful not to align all of the semantic roles, but only the vacant semantic role of the question with its corresponding semantic role in the answer passage. We have shown that it is also beneficial to score the event-bearing frames in the answer passages in the presence of frame redundancies in exhaustively annotated corpora.

Chapter 7

FrameNet Coverage and FrameNet-Based Answer Processing

The answer processing performance of a factoid QA system is dependent on the annotation accuracy of texts achieved by shallow semantic parsers as studied in Chapter 5. The task of annotation can be performed at different levels of correctness with different rates of coverage over lexical items (predicates). Therefore, the ongoing development of linguistic resources, which will increase the number of lexical items covered by each resource, may lead to increased performance of QA systems that employ these linguistic resources.

In all of the annotations carried out in our work, the set of semantic frames that are evoked by predicative targets is limited to the set of frames in the FrameNet 1.2 and 1.3 datasets. In this chapter, the effect of higher lexical coverage of FrameNet on the task of factoid answer processing is analysed. We show that higher lexical coverage results in more effective factoid answer processing. Attention is directed towards the importance of covering different part-of-speech predicates in FrameNet with the aim of improving the effectiveness of factoid answer processing[1]. This makes it possible to conduct future developments of FrameNet, and similar linguistic resources, in a way that they cover more of the parts-of-speech which play important roles in factoid natural language answer processing.

We first introduce the necessary concepts of lexical coverage in section 7.1. The analysis of the impact of lexical coverage on factoid answer processing will then be conducted in three steps: i) an inductive analysis of a random sample set of texts annotated with FrameNet elements (section 7.2), ii) a macro analysis of FrameNet datasets and their coverage rates over different part-of-speech lexical items (section 7.3), and iii) a QA-oriented analysis of different lexical coverage rates and answer processing performances (section 7.4).

[1] Some results of this study have been published in (Ofoghi, Yearwood, and Ma 2008*a*).

7.1 Linguistic Coverage

Since there are different features for each predicate, such as part-of-speech and sense (or semantic class), the coverage problem is considered to be a more complex issue than just including a predicate in the list of lexical items of a resource. Therefore, the current standing of linguistic resources in this respect can be measured in two lexical dimensions: i) *predicate coverage*, and ii) *sense coverage*.

7.1.1 Predicate Coverage

The first issue in lexical coverage of a linguistic resource relates to whether a predicate is included in the list of lexical items that are covered by semantic frames of that resource. If at least one single sense of a predicate with the same part-of-speech under consideration is covered in the resource, then the predicate is considered to be under the coverage of the resource.

For instance, the predicate "make.v" has many different senses such as "building", "cooking", "arriving", "causation", and "manufacturing". If any of these senses for the predicate "make.v" is included in the list of lexical items of a resource, then the predicate is considered as being covered by the resource, regardless of whether any of the other senses are included in the semantic frames of the resource. If all of the senses of a predicate are covered by the resource, then it is a full coverage; otherwise, it is a partial coverage. Therefore, all of the different semantic classes of a predicate participate in the task of measuring the coverage rate as a single item.

7.1.2 Sense Coverage

The second and more comprehensive way of measuring the rate of coverage of linguistic resources over lexical items considers not only the part-of-speech of predicates, but also the senses (or semantic classes) of them. This ensures that a broader linguistic feature of predicates is taken into account. This feature is concerned with a scenario or event in which a predicate plays the role of the main action occurring. As a result, the decision about a predicate to be covered by a lexical resource takes into consideration the contextual information of the predicate-containing sentences or paragraphs besides the individual features of the predicate in isolation.

Referring to the example predicate mentioned in section 7.1.1, the predicate "make.v" plays two totally different action roles, "cooking" and "manufacturing", in the example sentences below:

> Cooking → My mother makes excellent Iranian foods in a short amount of time. Manufacturing → Cars in many countries are made by well-established companies.

From this viewpoint, a lexical resource may include information on one sense of a predicate not covering the other semantic class(es). Therefore, each predicate along with its semantic class is considered as one item which participates in the task of measuring the coverage rate of a linguistic resource.

7.2 Naive Inductive Analysis of FrameNet Coverage

The FrameNet project is being developed on a frame basis instead of a predicate basis which slows down the task of covering English predicates. Each FrameNet frame can cover only a single sense of a predicate. Therefore, the frame-oriented development of FrameNet translates into sense-based progress of FrameNet coverage over predicates. This seems to be the reason for a relatively low rate of predicate coverage by FrameNet compared to other wide-coverage lexical semantic resources such as WordNet.

There is not much formal information about FrameNet coverage available; however, according to (Honnibal and Hawker 2005), the FrameNet 1.2 dataset covers only 64% of the tokens in the Penn Treebank and 26% of the token types. We conduct a naïve coverage analysis on parts of the text in the AQUAINT collection from which the answers for the TREC 2004 factoid questions are to be extracted. We explore a *random* list of top 10 passages retrieved for 10 factoid questions in the TREC 2004 track (100 passages in total). This analysis sheds some light on the proportions of coverage of different part-of-speech predicates in the FrameNet 1.3 dataset. Table 7.1 summarizes the statistical information of this sub-collection of AQUAINT.

The coverage analysis on this sub-collection measures the number of target predicates which could have been covered as FEEs, which evoke FrameNet semantic frames. From a statistical viewpoint, the minimum number of samples (predicates) required for analyzing the proportions at the confidence level 95% and margin of error 0.03 (desired precision 0.03) is 1068 (see Figure A.1 (Appendix A)). Therefore, even the total number of unique occurrences of predicates (1404) suffices for this analysis. Table 7.2 depicts the number of predicates which are not covered after the task of manual annotation using the FrameNet 1.3 dataset.

Table 7.1: The statistical information of a subset of the AQUAINT text collection on which an analysis of FrameNet coverage is conducted

Item	Number
Passages from AQUAINT documents	100
Sentences	233
Terms (all)	6006
Terms (unique)	1611
Predicates (all)	3567
Predicates (unique)	1404

The first column titled "Overall" in Table 7.2 shows the values acquired when taking into account all the sentences at once as a unique set. The "Avg." column, however, includes the values obtained as average values over 10 sets of sentences. The values in the unique not-covered predicate row (528 and 61.7) are not proportional as the uniqueness concept is interpreted differently with the different scopes for each column.

Table 7.2: All part-of-speech predicates not-covered after manual annotation with FN1.3

Item	Overall		Avg.	
	all	unique	all	unique
Predicates	1014	528	101.4	61.7
Normalized by sentences	4.351	2.266	4.348	2.711
Normalized by words	0.168	0.286	0.162	0.234
Normalized by predicates	0.284	0.376	0.274	0.325

The row "Normalized by words" in Table 7.2 shows that about 17% of the words (∼29% unique words) are not covered. It should be noted that these percentages are over the total number of the words in the set. In order to translate the values to a precise predicate coverage, it is required that the values be calculated as over the total number of predicates. The last row in Table 7.2 shows these numbers where almost 28% of the predicates (∼38% unique predicates) are not covered. This translates into ∼72% overall coverage for the predicates (∼62% coverage when considering the unique not-covered predicates). These results show both the predicate and sense coverage together.

Table 7.3: Verb predicates not-covered after manual annotation with FN1.3

Item	Overall		Avg.	
	all	unique	all	unique
Verbs	101	58	10.1	6.9
Normalized by sentences	0.433	0.248	0.430	0.314
Normalized by words	0.016	0.036	0.016	0.028
Normalized by predicates	0.028	0.043	0.027	0.037

Table 7.4: Noun predicates not-covered after manual annotation with FN1.3

Item	Overall		Avg.	
	all	unique	all	unique
Nouns	595	298	59.5	34
Normalized by sentences	2.553	1.278	2.601	1.502
Normalized by words	0.099	0.184	0.096	0.138
Normalized by predicates	0.166	0.223	0.163	0.184

A part-of-speech-based analysis of not-covered predicates is conducted to observe more detailed rates of lack of coverage over different part-of-speech predicates. The results are shown in Table 7.3 to Table 7.7 for verb, noun, adverb, adjective, and preposition predicates (leaving aside conjunctions and pronouns). By comparing the raw numbers of not-covered predicates, it can be seen that the majority of the not-covered predicates (in FrameNet) in our analysis collection are nouns. Figure 7.1 illustrates these raw measures. The other dominant predicates are adjectives and verbs while

Table 7.5: Adverb predicates not-covered after manual annotation with FN1.3

Item	Overall		Avg.	
	all	unique	all	unique
Adverbs	74	43	7.4	4.8
Normalized by sentences	0.317	0.184	0.305	0.208
Normalized by words	0.012	0.026	0.011	0.019
Normalized by predicates	0.020	0.032	0.019	0.025

Table 7.6: Adjective predicates not-covered after manual annotation with FN1.3

Item	Overall		Avg.	
	all	unique	all	unique
Adjectives	208	111	20.8	13.5
Normalized by sentences	0.892	0.476	0.854	0.573
Normalized by words	0.034	0.068	0.032	0.054
Normalized by predicates	0.058	0.083	0.054	0.072

not many adverbs and prepositions are among not-covered predicates. These measures, however, do not indicate what percentages of different part-of-speech predicates are covered/not-covered in the analysis collection.

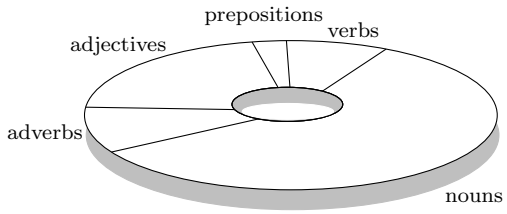

Figure 7.1: Raw figures of *all* different part-of-speech predicates (in our analysis sub-collection) not-covered in FN1.3

A final analysis on the same collection is performed to observe the proportions of different part-of-speech predicates that are covered/not-covered. The results of this analysis, summarized in Table 7.8 and Table 7.9, show these measures with respect to *all* and *unique* occurrences respectively. In all occurrences, after preposition predicates which are ∼96% covered, verb predicates are shown to have been covered more than the other predicates (∼78%). In the unique occurrences, verb predicates are covered more than the other predicates (∼73%). Preposition (∼69%) and noun (∼65%) predicates come after verbs. Overall, it can be seen that the coverage rates in FrameNet 1.3, for different part-of-speech predicates, tend to be low.

Table 7.7: Preposition predicates not-covered after manual annotation with FN1.3

Item	Overall		Avg.	
	all	unique	all	unique
Prepositions	23	13	2.3	1.6
Normalized by sentences	0.098	0.055	0.103	0.073
Normalized by words	0.003	0.008	0.003	0.006
Normalized by predicates	0.006	0.009	0.006	0.008

Table 7.8: Part-of-speech-distinguished analysis of predicate coverage in our analysis sub-collection using FN1.3 - *all* occurrences

POS	#Total	Covered		Not-covered	
		#	%	#	%
Verb	465	364	78.279	101	21.721
Noun	1947	1352	69.440	595	30.560
Adverb	165	91	55.151	74	44.849
Adjective	361	153	42.382	208	57.618
Preposition	598	575	96.153	23	3.847

7.3 FrameNet Statistics

Before conducting experiments on different versions of FrameNet datasets, some statistical information about the two versions of FrameNet are required so that the QA performances with the two FrameNet datasets can be better analyzed. With this macro analysis of FrameNet, Table 7.10 summarizes information on the raw numbers of total Lexical Units (LUs), verbs, nouns, adjectives, adverbs, and prepositions. It also shows the total number of frames and FEs in the two datasets. The measures are calculated by a software program that we have implemented to access the FrameNet XML datasets.

In order to observe the growth ratio on each item in Table 7.10, the formula in Equation 7.1 is used where $population_i^1$ and $population_i^2$ indicate the total number of each item i in the FrameNet 1.2 and FrameNet 1.3 datasets respectively and gr_i stands for the growth ratio for each item i.

$$gr_i = \frac{population_i^2 - population_i^1}{population_i^1} \times 100 \tag{7.1}$$

From a LU (predicate) coverage point of view, prepositions have the highest ratio of growth with ∼26%. After prepositions, the growth ratio of adverbs and verbs are higher than the other part-of-speech predicates.

From a FrameNet elements (frames and FEs) viewpoint, the growth ratios, however, are more promising. This is due to the fact that the work on progressing FrameNet is conducted on a frame basis.

Table 7.9: Part-of-speech-distinguished analysis of predicate coverage in our analysis sub-collection using FN1.3 - *unique* occurrences

POS	#Total	Covered		Not-covered	
		#	%	#	%
Verb	216	158	73.148	58	26.852
Noun	875	577	65.942	298	34.058
Adverb	69	26	37.681	43	62.319
Adjective	197	86	43.654	111	56.346
Preposition	42	29	69.047	13	30.953

Table 7.10: Statistical information of the FN1.2 and 1.3 datasets

Item	LUs	Verbs	Nouns	Adj.	Adv.	Prep.	Frames	FEs
FN1.2 dataset	8755	3424	3673	1536	39	72	609	4908
FN1.3 dataset	9454	3891	3730	1680	49	91	795	7124
Growth ratio (%)	7.984	13.639	1.551	9.375	25.641	26.388	30.541	45.150

In Table 7.11, we summarize the results of our different FrameNet-based analysis on coverage and growth ratio for each part-of-speech predicate. The current coverage ratios are those induced by our naïve analysis in section 7.2 for unique occurrences of predicates according to the FrameNet 1.3 dataset. The growth ratios are actual measures acquired by Equation 7.1. The next release coverage ratios are predicated by assuming that the growth ratios for each item will remain constant until the next release of FrameNet dataset.

The predicated coverage ratios are calculated by Equation 7.2 where pcr_i stands for predicate coverage ratio, ccr_i indicates the current coverage ratio, and gr_i shows the growth ratio for each item i.

$$pcr_i = \frac{(ccr_i \times gr_i)}{100} + ccr_i \tag{7.2}$$

From the measures in Table 7.11, by assuming that the growth ratios will remain the same, it can be seen that the progress in covering prepositions and verbs is better than the other part-of-speech predicates. In general, noun predicates still need some more effort where the task of covering adverbs and adjectives seems to be crucial which requires more work. These conclusions are drawn in a general sense of FrameNet coverage. In terms of factoid QA, however, we need more specifically related experiments, as explained in the next section, to understand the importance of covering lexical units in order to more effectively answer factoid questions.

Table 7.11: Coverage and growth ratio of different part-of-speech predicates in FN1.3

POS	Current coverage ratio (%)	Growth ratio (%)	Predicted next release coverage ratio (%)
Verb	73.148	13.639	83.124
Noun	65.942	1.551	66.964
Adverb	37.681	25.641	47.342
Adjective	43.654	9.375	47.746
Preposition	69.047	26.388	87.267

7.4 Practical Analysis of FrameNet Coverage and Factoid Answer Processing

To analyse the impact of FrameNet coverage over lexical units on factoid answer processing performance, we carry out two sets of experiments on the two TREC 2004 and TREC 2006 factoid question sets (see details of the data in Chapter 3). We consider two facets in these experiments:

i) The final set of frames used for annotating (or manually correcting the annotations of) questions and passages, and
ii) The training process of shallow semantic parsers which annotate the text of questions and passages with FrameNet elements.

In experimenting on the TREC 2004 factoid question set, we run our experimental QA system, explained in Chapter 3, on the annotated questions and retrieved passages which are firstly annotated by the SHALMANESER parser trained with the FrameNet 1.2 dataset. To see the impact of different FrameNet coverage rates on the factoid answer processing task, we run the QA system on:

- The annotated questions and passages (manually corrected) with the frames in the FrameNet 1.2 dataset,
- The annotated questions and passages (manually corrected) with the frames in the FrameNet 1.3 dataset,
- The annotated questions and passages with SHALMANESER trained with the FrameNet 1.2 dataset without any manual correction, and
- The annotated questions and passages with SHALMANESER trained with the FrameNet 1.3 dataset without any manual correction.

In the case of the experiment on the TREC 2006 dataset, since there is no manual correction performed on the annotations, we run the QA system on:

- The annotated questions and passages with SHALMANESER trained with the FrameNet 1.2 dataset, and

- The annotated questions and passages with SHALMANESER trained with the FrameNet 1.3 dataset.

7.4.1 Experimental Results

Details of the experimental setup, data, required software modules, and tools can be found in Chapter 3. In this section, the results obtained for the two above-mentioned types of FrameNet coverage analysis are shown. In these experiments, the FSB-only setting of answer processing is used for retrieving answers and the frame semantic-based model takes the FSFE-NFS method of FrameNet-based answer processing.

Table 7.12 contains the results of the experimental QA runs on 75 TREC 2004 factoid questions with two different sets of frames (from FN1.2 and FN1.3) in the final annotated questions and passages. The statistical significance of the differences between the results obtained using the two different FrameNet framesets is quantified by the calculation of p-values after paired t-tests (see section 3.2.7). Table 7.13 summarizes the results on the same dataset without any manual correction of annotations. The annotations in Table 7.13 are based on two versions of SHALMANESER trained with the two different FrameNet datasets 1.2 and 1.3.

Table 7.14 shows the results of the QA runs on 176 TREC 2006 factoid questions. The results are based on two instances of SHALMANESER where its learning classifier for frame evocation - the FRED classifier - is trained with two different versions of FrameNet dataset.

Table 7.12: QA runs with different frame sets in different FrameNet datasets on 75 TREC 2004 factoid questions - values with † are statistically significant ($p < 0.05$)

FrameNet dataset	*mrr*	
	strict	lenient
FN1.2	0.560	0.613
FN1.3	0.587 *p=0.079*	0.640 *p=0.079*

Table 7.13: QA runs with different FrameNet datasets used for training SHALMANESER on 75 TREC 2004 factoid questions - values with † are statistically significant ($p < 0.05$)

FrameNet dataset	*mrr*	
	strict	lenient
FN1.2	0.000	0.000
FN1.3	0.013 *p=0.160*	0.013 *p=0.160*

Table 7.14: QA runs with different FrameNet datasets used for training SHALMANESER on 176 TREC 2006 factoid questions - values with † are statistically significant ($p < 0.05$)

FrameNet dataset	*mrr*	
	strict	lenient
FN1.2	0.011	0.017
FN1.3	0.006 *p=0.159*	0.011 *p=0.159*

7.4.2 Discussion

The first observation from the experimental results in Table 7.12, Table 7.13, and Table 7.14 is that in an effectively annotated environment, there is a higher chance of retrieving more correct factoid answers for the frame semantic-based answer processing module as the coverage ratio of predicates in FrameNet grow. In other words, the higher coverage rate of predicates in FrameNet along with an accurate annotation task - such as that performed in the TREC 2004 dataset - results in a higher factoid answer processing performance as would intuitively be expected.

The improvement in the QA performances - in terms of *mrr* values - with different lexical coverage rates in FrameNet, however, is not statistically significant at this time. With respect to the growth ratios of covering more predicates in FrameNet 1.3 compared to FrameNet 1.2, this is normally expected. For a more significant progress in factoid *mrr* measures, in a comprehensively annotated environment, it is necessary to cover more predicates and their senses in the next FrameNet versions.

After our analysis in section 7.2, it is shown that noun predicates are covered less than all other part-of-speech predicates. Intuitively, it is expected that in finding answers to factoid questions, verb and noun predicates play more important roles. This is because the main actions of the question events are more associated with the verbs and nouns in the questions. The results obtained after the experiments on verb-only frames (SHAL-VF) in Chapter 5 show the importance of verb predicates. Furthermore, the induced growth ratio in terms of verbs (13.639%) in FrameNet is more promising than that of nouns (1.551%). These facts and the conclusion that can be drawn are summarized as below:

- Fact: Nouns are covered poorly in FrameNet.
- Fact: Verbs and nouns play important roles in answering factoid questions.
- Fact: The current growth ratio of nouns is not promising.
- Conclusion: The work on covering a greater number of nouns in FrameNet is crucial at this stage to balance coverage rates in the next releases of FrameNet. This can increase the potential for factoid QA systems to extract a greater number of correct answers.

This conclusion is drawn based on the growth model presented in section 7.3 which is not a perfect model. We are aware of the following issues that may affect this conclusion:

- The growth ratios and predicted coverage rates shown in Table 7.11 are less likely to remain

constant for prepositions, adverbs, and verbs. This is because if the growth ratios remain the same, then the coverage rates for these predicates will be 100% in the near future. However, the growth ratios for nouns and adjectives are more likely to remain the same.

- New predicates to be covered in FrameNet are more likely to be those predicates which are used less frequently in the language and therefore, are less likely to occur in questions and answer passages.
- With the continual generation of noun phrases and also proper nouns, it is very hard for FrameNet's noun coverage to be balanced with that of its other part-of-speech predicates in the near future.

By focusing on the results on the TREC 2006 dataset in Table 7.14, it is inferable that in a sparsely annotated text collection, a higher predicate coverage may even damage the QA performance. This can also be inferred from Table 7.13 on the automatically annotated TREC 2004 dataset without manual corrections. The reason for this situation is that in a sparse and inaccurate annotation environment, resulting from an inaccurate automated parser, there is further possibility for extracting wrong answers by a greater number of wrongly assigned frames and FEs (negative frame redundancy). Once again, this emphasizes the importance of semantic class identification and semantic role labeling with respect to FrameNet frames and FEs, which should be combined with a high predicate and sense coverage.

7.5 Summary

The coverage rate of FrameNet over different part-of-speech predicates has been analysed in an inductive naïve method and a macro analysis of FrameNet in this chapter. It has been shown that the coverage rates of different part-of-speech predicates in FrameNet 1.3 are not very high. Also, the growth ratios of covering more predicates from the FrameNet 1.2 dataset to the FrameNet 1.3 dataset are not very high for different part-of-speech lexical units.

With this information, the effect of different existing coverage rates of predicates in the two FrameNet datasets 1.2 and 1.3 has been analysed in factoid frame semantic-based answer processing. The results have shown that with the existing growth ratios the improvement over the QA performances is not statistically significant, although there is some improvement in a comprehensively annotated environment. The work on covering more noun predicates has been inferred to be most crucial in elevating the factoid answer processing performance in the future versions of FrameNet as the growth ratio for the only other important part-of-speech predicates (verbs) is relatively high in FrameNet at this stage. The higher FrameNet coverage without having a precise annotation task has been shown to have no certain positive impact on finding a greater number of correct factoid answers.

Chapter 8

Fusion of FrameNet-Based Answer Processing and Non-Semantic Approaches

With different answer processing models and algorithms, each of which performs well in identification of certain types of factoid answers, it is important to make use of a combination of methods. In this chapter, we test the overall effectiveness of factoid answer processing, using a combination of two answer processing models, in a range of different textual situations. We propose two different methods of fusing the results of a frame semantic-based answer processing model with those of a non-semantic entity-based model[1]. The first method uses answer scores for merging two answer lists while the second technique utilizes the ranks of answers regardless of their actual scores. Further analysis are conducted on the score-based technique as it is shown to generally perform more effectively. This analysis includes a tuning process for the convex linear parameter of the fusion function and an investigation into the correct answer coverage by this fusion technique.

8.1 Motivation

With current state-of-the-art semantic parsers, there are still a few problems which interfere with the performance of a frame semantic-based model when used solely in the task of factoid answer processing for QA. Some of the issues that challenge the frame semantic-based model include:

- The current incomplete coverage of FrameNet over different part-of-speech predicates as explained in Chapter 7.
- The non-predicate-argument structure of answer-containing text spans in some cases which results in no frame evocation from the FrameNet dataset.
- Frame mismatches between a given question and its answer passages due to different scenarios or dissimilar contextual backgrounds. This has been explained in Chapter 5.

[1] Parts of the material in this chapter have been published in (Ofoghi, Yearwood, and Ma 2009).

While the first problem can be alleviated by accessing more complete versions of FrameNet over time with wider coverage rates over lexical units, the two other issues strongly suggest the usage of a semantic approach in conjunction with a non-semantic method that does not depend on the syntactico-semantic structure of question and answer passage texts. As a result, we implement a named entity-based model of answer processing in our experimental QA system (see Chapter 3 for details) the results of which are fused with those of the frame semantic-based model.

From a fusion viewpoint, the two models have to be automated. The entity-based model is a fully automated answer processing model. The frame semantic-based model, on the other hand, processes texts which are annotated automatically and improved manually. Overall, this setting suggests a valid fusion exercise since the frame semantic-based model also carries out the task of answer processing in an automated fashion.

These two different models have different characteristics which yield different abilities in factoid answer identification; therefore, they can cover different sets of correctly answered questions. Since each answer processing model retrieves a list of answer candidates per question, the task of fusing the answer lists of each model for each question is a crucial step towards making use of the two answer processing models in an effective way. The effectiveness of a fusion method is related to the *negative impact* that each answer processing model can impose on the extraction performance of the other model when they are combined with each other and there is only a single answer accepted as a final response to the questions. The negative impact refers to the situations where incorrect answer candidates of a model are wrongly ranked as the first answers in the merged list of answer candidates. This prevents the QA system from reporting the correct answers that are extracted by the other model. Therefore, it is necessary for an answer list fusion method to minimize this impact of the answer processing models on each other to provide a useful synergy of different answer processing models.

8.2 Answer List Fusion Methods

We use fusion techniques to merge answer lists retrieved by each answer processing model. The main parameters that our fusion techniques consider are the characteristics of a a certain answer in its containing list which include:

- **The answer score:** which expresses information about how much an answer processing model is confident about the answer to be a real and correct answer to given question. The answer scoring process in each answer processing model is carried out in a different way. In the frame semantic-based model, there are a number of different techniques explained in Chapter 5. In the entity-based model, however, the answer candidates are scored according to the similarity measure between their containing passages and the questions.

- **The answer rank:** which shows how an answer processing model rates a certain answer among a possible set of answers.
- **The answer redundancy:** which emphasizes the confidence and persistence of an answer processing model in retrieving an answer.

We introduce two methods of answer list fusion based on the three main characteristics of answers. The first method merges the answer lists of the two answer processing models with respect to the answer ranks. The second method, in contrast, takes into account the answer scores that are calculated by each model. Both methods consider the answer redundancy as a positive point in selecting an answer.

8.2.1 Rank-Based Fusion

The rank-based fusion technique that we develop is based on the ranks of answer candidates in each list of answers retrieved by each answer processing model. This method does not make use of the scores of answer candidates in each individual answer list in the fusion process. Instead, it focuses on the rank of answers and their redundancies in the retrieved answer lists. The main reason behind this type of fusion is to ignore different answer scoring schemas developed in the two answer processing methods so that they are treated as if they were retrieved by a single answer processing method.

Before explaining the different steps of this fusion method, it is necessary to define three required specific concepts:

- **Answer pair:** is a pair of answer candidates which contains the top-ranked answer candidate of each answer list.
- **Internal redundancy:** is the frequency of the occurrence of an answer in its original container list. This is used to calculate the probability of an answer candidate in the container answer list extracted by a certain answer processing model.
- **External redundancy:** refers to the frequency of the occurrence of an answer in the list of answers retrieved by the other answer processing model.

The main procedure of fusing answer lists in the rank-based method involves seven steps:

i) The scored answer candidates extracted by each model are stored in a sorted list of the maximum five answer candidates. Sorting of these answer candidates is carried out according to their scores and redundancies (multiplying answer scores by the probabilities of their occurrence in the answer lists).

ii) The first answer candidate with the highest score from each answer list is taken into an answer pair.

iii) The internal redundancy of each single answer in the answer pair is calculated in its containing list.

iv) The external redundancy of each single answer in the answer pair is considered to calculate the external reciprocal rank of the answer in the list of answers extracted by the other model.

v) Having the internal redundancy (probability) and external reciprocal rank calculated, the rank-based value of each single answer in the answer pair is measured using Equation 8.1, where f_{apm} refers to the rank-based value of the answer ans_{apm} retrieved by the answer processing model apm, err_i stands for the external reciprocal rank i and p_{int} refers to the internal probability of the same answer. Since there may be n occurrences of an answer at different positions in the list of retrieved answers by the other model, we consider the summation of external reciprocal ranks. The parameter $\varepsilon(> 0)$ is used to avoid null values for answers in case there is no inter-list support. In our work we set $\varepsilon = 0.01$.

$$f_{apm}(ans_{apm}) = \sum_{i=1}^{n}(err_i + \varepsilon) \times p_{int} \tag{8.1}$$

vi) The rank-based value of each single answer in the answer pair is scaled using the convex linear function of Equation 8.2 where α is the convex parameter ($0 \leq \alpha \leq 1$).

$$\alpha \times f_{fsb}(ans_{fsb}) + (1 - \alpha) \times f_{enb}(ans_{enb}) \tag{8.2}$$

vii) The single answer from the answer pair with the highest rank-based value is selected as the final answer.

Example 8.1-

Retrieved passages for the question "What years did she accompany Lewis and Clark on their expedition?" (Q44.2 in the TREC 2004 QA track) are submitted to the answer processing module of our experimental QA system with the two models, entity-based and frame semantic-based answer processing models. The answer lists retrieved by these two models are shown in Table 8.1. The answers in each list are ranked by the answer scores that each individual model calculates for the answers. These scores are ignored in the process of fusing the results; therefore, they are not shown in Table 8.1.

Table 8.1: The answer lists of the two answer processing models for the question Q44.2 in the TREC 2004 QA track retrieved by our experimental QA system

Answer rank	ENB answer	FSB answer
1	16-year-old	in 1804-06
2	16-year-old	in 1804-06
3	16-year-old	Null
4	1804-06	Null
5	1805,	Null

The answer pair <"16-year-old", "in 1804-06"> is constructed by selecting the first answers from each list. The rank-based values of single answers "16-year-old" and "in 1804-06" are calculated according to the steps defined above. This results in the pair <0.006, 0.004> for the rank-based values corresponding to the answer pair <"16-year-old", "in 1804-06">[2]. By comparing the two rank-based values 0.006 and 0.004, the answer candidate "16-year-old" is selected as the answer to be reported which corresponds to the rank-based value 0.006.

In finding internal and external redundancies for the single answer "in 1804-06", the string "1804-06" is not considered to be a redundant answer by a strict matching procedure that is performed in our experiments.

A post-processing task of the answers removes any prepositions such as "in", "for", "from", "by", "at", and "on" at the beginning of the final answer strings.

8.2.2 Score-Based Fusion

The methodology of fusing answer lists in the score-based technique, in contrast with the rank-based method, relies on answer scores calculated by each answer processing model. In this method, the ranks of answers in each answer list are not taken into consideration. The main emphasis of this fusion technique is on the answer scores and answer redundancies to change the answer scores calculated by each answer processing model. The score-based technique of fusing answer lists consists of five steps as below:

i) The scored answer candidates extracted by each model are stored in a sorted list of the maximum five answer candidates (similar to the first step in rank-based fusion).

ii) A single list of answers is constructed by concatenating the two answer lists retrieved by each answer processing model.

iii) The answer scores are changed according to the convex linear function shown in Equation 8.3 where $S_{ans_{fsb}}$ and $S_{ans_{enb}}$ are the scores of the answers retrieved by the frame semantic-based and entity-based answer processing models respectively and α is the convex parameter which differentiates the emphasis on the answers retrieved by the different models.

$$\alpha \times S_{ans_{fsb}} + (1 - \alpha) \times S_{ans_{enb}} \tag{8.3}$$

iv) In the single list of answers, where the answer scores are combined using the convex linear function, internal answer redundancies (the probabilities of each answer in the single answer list) are used to change the answer scores. This is done by multiplying the answer scores by the probabilities.

v) The single list of answers is sorted according to the final answer scores and the top answer is reported as the final answer.

[2] f_{enb}("16-year-old")$= (0 + 0.01) \times 0.6 = 0.006$ and f_{fsb}("in 1804-06")$= (0 + 0.01) \times 0.4 = 0.004$

The value of the convex parameter α in the convex linear function of Equation 8.3 plays an important role in optimally emphasizing the answers of the two answer processing models. A more detailed analysis of this parameter will be conducted in section 8.3.1.

Example 8.2-

The same question as in Example 8.1 (Q44.2 in the TREC 2004 QA track) is submitted to the passage retrieval module of our experimental QA system and the top 10 retrieved passages are submitted to the answer processing module with the two entity-based and frame semantic-based models. The top five answers of each model are shown in Table 8.2 with their scores.

Table 8.2: The answer lists and answer scores obtained by the two answer processing models for the question Q44.2 in the TREC 2004 QA track retrieved by our experimental QA system

Answer rank	ENB		FSB	
	answer	score	answer	score
1	16-year-old	0.175	in 1804-06	0.700
2	16-year-old	0.175	in 1804-06	0.514
3	16-year-old	0.128	Null	-
4	1804-06	0.116	Null	-
5	1805,	0.090	Null	-

The score-based fusion method constructs a single list of the answers and their scores and changes the scores according to the redundancies of the answers. Table 8.3 includes the sorted list of answers with their scores before and after redundancy-based changes are applied. The final scores in Table 8.3 are obtained after using the linear convex parameter $\alpha = 0.5$ as well. Similar to the redundancy processing procedure in the rank-based method, the answers are considered to be redundant if they strictly match.

Having this single answer list constructed, the first answer "in 1804-06" with the highest score 0.285 is selected as the final answer. This answer is then post-processed to remove the preposition "in" from the beginning and the final string "1804-06" is reported.

8.2.3 Experimental Results

The two fusion methods (rank-based and score-based) are implemented in our experimental QA system. The answer processing module in these experiments works on the basis of the Merged (FSB-ENB-fused) strategy explained in Chapter 3. An ENB-only setting of answer processing is considered as the baseline (BL) system. The two fusion methods are applied to the answers which are extracted by the two answer processing models described in the same chapter (the entity-based and frame semantic-based models).

We run the experimental QA system on both the TREC 2004 and the TREC 2006 datasets. The

Table 8.3: The single answer list and answer scores after score-based merging for the question Q44.2 in the TREC 2004 QA track retrieved by our experimental QA system

Answer rank	ENB-FSB answer	Score calculated by model	Probability in the single list	Final score
1	in 1804-06	0.700	2/7=0.285	0.100
2	in 1804-06	0.514	2/7=0.285	0.073
3	16-year-old	0.175	3/7=0.428	0.037
4	16-year-old	0.175	3/7=0.428	0.037
5	16-year-old	0.128	3/7=0.428	0.027
6	1804-06	0.116	1/7=0.142	0.008
7	1805,	0.090	1/7=0.142	0.006

frame semantic-based model extracts answers from annotated text with the SHALMANESER parser which has been trained using the FrameNet 1.2 data. Manual correction of annotated passages and questions in the TREC 2004 dataset is performed with the frames and FEs in the FrameNet 1.3 dataset which has a higher rate of coverage over English predicates. The answer processing procedure in the frame semantic-based model takes the different frame semantic alignment methods the details of which have been given in Chapter 6.

In using the convex linear function in the answer fusion methods in our experiments we set $\alpha = 0.5$. For more analysis on the effect of different α values, we will conduct another study in section 8.3.1. Table 8.4 to Table 8.7 summarize the results obtained on the two datasets TREC 2004 and TREC 2006 for the rank-based and score-based fusion methods. There is only a single annotation level in the TREC 2006 factoid question set - the SHALMANESER (SHAL) level - as there was no manual correction of the annotations related to our 2006 experiments due to time and cost limitations. In the case of the 2004 experiments, however, the results are reported with respect to all annotation levels described in Chapter 5.

8.2.4 Discussion

A few observations can be made from the QA results obtained in section 8.2.3. First, the rank-based fusion method results in improvements over the baseline performance in both TREC 2004 and 2006 datasets. The score-based method, however, provides improvement over the baseline answer processing performance only in the TREC 2004 dataset where there are manual corrections on the annotations. This means that a poor performing frame semantic-based model (in a sparsely annotated environment) can have more negative effects on the performance of the entity-based model when using the score-based fusion method. In general, the answer candidates from the frame semantic-based model get higher scores than those of the entity-based model for many questions in the score-based fusion method. This is because frame semantic-based answer candidates are scored highly since they are extracted from highly ranked passages and their scores are, in some methods

Table 8.4: QA runs on 75 TREC 2004 factoid questions with the rank-based fusion method, bold numbers show maximum values in each column

QA and parsing level	FSB method	*mrr*					
		strict			lenient		
		FSB	ENB	fused	FSB	ENB	fused
Baseline answer processing (BL)	N/A	N/A	**0.400**	0.400	N/A	**0.413**	0.413
BL + SHAL	CFFE	0.000	0.387	0.387	0.000	0.400	0.400
	FSFE-NFS	0.000	0.387	0.387	0.000	0.400	0.400
	FSFE-FS	0.000	0.387	0.387	0.000	0.400	0.400
	FE-NFES	0.000	0.373	0.373	0.000	0.387	0.387
	FE-FES	0.000	0.373	0.373	0.000	0.387	0.387
BL + SHAL-VF	CFFE	0.080	0.333	0.413	0.093	0.347	0.440
	FSFE-NFS	0.133	0.307	0.440	0.160	0.320	0.480
	FSFE-FS	0.133	0.307	0.440	0.160	0.320	0.480
	FE-NFES	0.040	0.360	0.400	0.053	0.373	0.427
	FE-FES	0.040	0.360	0.400	0.053	0.373	0.427
BL + SHAL-AF	CFFE	0.107	0.320	0.427	0.133	0.333	0.467
	FSFE-NFS	0.160	0.280	0.440	0.200	0.293	0.493
	FSFE-FS	0.160	0.280	0.440	0.200	0.293	0.493
	FE-NFES	0.080	0.267	0.347	0.107	0.280	0.387
	FE-FES	0.080	0.267	0.347	0.107	0.280	0.387
BL + SHAL-HL	CFFE	0.160	0.280	0.440	0.200	0.293	0.493
	FSFE-NFS	**0.240**	0.253	**0.493**	**0.280**	0.267	**0.547**
	FSFE-FS	**0.240**	0.253	**0.493**	**0.280**	0.267	**0.547**
	FE-NFES	0.160	0.240	0.400	0.200	0.253	0.453
	FE-FES	0.173	0.240	0.413	0.200	0.253	0.453

(FSFE-FS and FE-FES), summed with additional values (see Chapter 6). Therefore, many incorrect frame semantic-based answers may have high scores and dominate. Lower overall answer processing performances can be achieved as a result of incorrect frame semantic-based answers dominating the final answer list.

Second, the score-based fusion method performs slightly better than the rank-based method in the TREC 2004 factoid question set using all frame semantic-based answer processing techniques except for the CFFE technique. In the TREC 2006 dataset, however, the rank-based fusion method outperforms the score-based method with respect to all frame semantic-based answer processing techniques. The reason for this is the smaller number of answer candidates extracted using the CFFE's complete FE matching procedure. With fewer lowly scored answer candidates from CFFE, which cannot be ranked at the top of the answer list in the score-based fusion technique, the rank-based technique of fusing CFFE with the entity-based model outperforms the same combination of answer processing models using the score-based fusion technique. This is also due to the fact that the fewer answers from CFFE results in a low answer redundancy in the list of extracted answer candidate by the frame semantic-based model. This causes less negative impact over the performance of the entity-based model. Therefore, the overall answer processing performance is higher compared to the score-based technique.

Third, in the score-based fusion method, there is more opportunity for the frame semantic-based

Table 8.5: QA runs on 176 TREC 2006 factoid questions with the rank-based fusion method

QA and parsing level	FSB method	*mrr*					
		strict			lenient		
		FSB	ENB	fused	FSB	ENB	fused
Baseline answer processing (BL)	N/A	N/A	0.097	0.097	N/A	0.142	0.142
BL + SHAL	CFFE	0.006	0.102	0.108	0.006	0.148	0.153
	FSFE-NFS	0.011	0.102	0.114	0.011	0.148	0.159
	FSFE-FS	0.011	0.102	0.114	0.011	0.148	0.159
	FE-NFES	0.011	0.102	0.114	0.011	0.148	0.159
	FE-FES	0.011	0.102	0.114	0.011	0.148	0.159

answer processing model to return more correct answers compared to the entity-based model. This is again due to the high scores of answer candidates extracted by the frame semantic-based model (explained above). This is observed to be true in the results on both the TREC 2004 and 2006 datasets.

Fourth, the rank-based fusion method generally makes the entity-based model outperform the frame semantic-based model due to the higher answer redundancies for those answers extracted by the entity-based model. Since the rank-based method works on the basis of answer redundancies, this observation can be further explained as follows:

i) The answers obtained using the entity-based model are more internally redundant within the entity-based answer list, and
ii) The answers extracted by the entity-based model are more likely to be also retrieved by the frame semantic-model. This results in higher external redundancies for the entity-based answers. However, the answers which are extracted by the frame semantic-based model are more unique to this model so that they are not likely to be also retrieved by the entity-based model.

Fifth, in a completely and comprehensively annotated environment the frame semantic-based model has a greater chance of obtaining more correct answers than the entity-based model. The main reason is the reliance of the score-based method on the frame semantic-based model which is more effective at higher levels of frame semantic-based text annotation. The results obtained in the TREC 2004 and TREC 2006 datasets confirm this argument where in the case of the former (with manual annotation corrections) the score-based method outperforms the rank-based fusion technique (except for when using CFFE) and in the latter (with no manual annotation) the scenario is vice versa.

Sixth, the two answer processing models may negatively affect the answer processing performance of each other. In section 8.3, the score-based technique, which overall outperforms the rank-based technique in comprehensively annotated texts, is further analysed to see how it is possible to reduce the negative impact for the answer processing models on each other and achieve high overall answer processing *mrr*s.

Table 8.6: QA runs on 75 TREC 2004 factoid questions with the score-based fusion method, bold numbers show maximum values in each column

QA and parsing level	FSB method	*mrr*					
		strict			lenient		
		FSB	ENB	fused	FSB	ENB	fused
Baseline answer processing (BL)	N/A	N/A	**0.400**	0.400	N/A	**0.413**	0.413
BL + SHAL	CFFE	0.000	0.347	0.347	0.000	0.360	0.360
	FSFE-NFS	0.000	0.347	0.347	0.000	0.360	0.360
	FSFE-FS	0.000	0.333	0.333	0.000	0.347	0.347
	FE-NFES	0.000	0.333	0.333	0.000	0.347	0.347
	FE-FES	0.000	0.320	0.320	0.000	0.333	0.333
BL + SHAL-VF	CFFE	0.093	0.307	0.400	0.107	0.320	0.427
	FSFE-NFS	0.160	0.253	0.413	0.187	0.267	0.453
	FSFE-FS	0.213	0.213	0.427	0.240	0.227	0.467
	FE-NFES	0.067	0.320	0.387	0.080	0.333	0.413
	FE-FES	0.160	0.200	0.360	0.187	0.213	0.400
BL + SHAL-AF	CFFE	0.147	0.280	0.427	0.173	0.293	0.467
	FSFE-NFS	0.227	0.213	0.440	0.267	0.227	0.493
	FSFE-FS	0.307	0.147	0.453	0.347	0.160	0.507
	FE-NFES	0.120	0.253	0.373	0.160	0.267	0.427
	FE-FES	0.240	0.133	0.373	0.280	0.147	0.427
BL + SHAL-HL	CFFE	0.200	0.213	0.413	0.253	0.227	0.480
	FSFE-NFS	0.413	0.107	0.520	0.467	0.120	0.587
	FSFE-FS	**0.600**	0.000	**0.600**	**0.653**	0.000	**0.653**
	FE-NFES	0.227	0.213	0.440	0.267	0.227	0.493
	FE-FES	0.427	0.000	0.427	0.467	0.000	0.467

8.3 Further Analysis on Score-Based Fusion

The score-based fusion method to merge the lists of answer candidates retrieved by the two answer processing models (frame semantic-based and entity-based) can be further developed. This development may result in a lower negative impact for the two answer processing models on the performance of each other. The main reasons for selecting the score-based fusion method for further analysis are:

- Generally, the score-based fusion method slightly outperforms the rank-based method in our experiments on factoid answer processing in comprehensively annotated environments, and
- The score-based fusion method is more dependent on the frame semantic-based answer processing model.

Therefore, the score-based technique is further investigated in the following two sub-sections to answer two questions:

i) What is the effect of different values for the convex linear parameter α in the overall performance of factoid answer processing using the score-based fusion technique?

ii) What is the correct answer coverage rate of the merged frame semantic-based and entity-based answer processing technique with respect to certain settings of score-based fusion?

Table 8.7: QA runs on 176 TREC 2006 factoid questions with the score-based fusion method

QA and parsing level	FSB method	*mrr*					
		strict			lenient		
		FSB	ENB	fused	FSB	ENB	fused
Baseline answer processing (BL)	N/A	N/A	0.097	0.097	N/A	0.142	0.142
BL + SHAL	CFFE	0.006	0.085	0.091	0.011	0.125	0.136
	FSFE-NFS	0.011	0.085	0.097	0.017	0.125	0.142
	FSFE-FS	0.011	0.085	0.097	0.017	0.125	0.142
	FE-NFES	0.011	0.085	0.097	0.017	0.125	0.142
	FE-FES	0.011	0.085	0.097	0.017	0.119	0.136

8.3.1 Tuning Fusion Parameter

The convex linear function shown in Equation 8.3 for controlling the emphasis of the score-based fusion method towards the different answer processing models plays an important role in the overall answer processing performance. By setting the convex linear parameter α to different values, it is easily possible to change the emphasis of the fusion method and consequently affect the overall answer processing performance.

To analyse the impact of this convex linear parameter, a set of QA runs is carried out using the score-based fusion method to merge the lists of answer candidates retrieved by the entity-based and frame semantic-based answer processing models and select final answers. A number of distinct values for α in the range of [0.0,1.0] with the step value 0.05 are selected to be applied in these QA runs over the TREC 2004 factoid question set. The frame semantic-based model, in this experiment, works with the FSFE-NFS and FSFE-FS methods as they have already shown the highest answer processing performances among all of the frame semantic-based answer processing techniques. The automated annotation is performed using SHALMANESER trained on FrameNet 1.2. The manual correction of annotation is based on FrameNet 1.3.

The FSFE-NFS method of frame semantic-based answer processing is selected as it does not perform any extra answer scoring procedure than the passage-based scoring which scores each answer with the score of its containing passage. This results in an equal answer scoring method with no methodological bias towards any of the models.

The FSFE-FS method is used to observe differences that may occur while more comprehensively scoring answer candidates of the frame semantic-based answer processing model compared to the answers which are retrieved by the entity-based model.

The results of applying different α values to the score-based fusion method with the FSFE-NFS and FSFE-FS methods in the frame semantic-based answer processing model are shown in Table 8.8 and Table 8.9 respectively. Figure 8.1 and Figure 8.2 graphically display the trends of the factoid answer processing *mrr*s in the strict evaluation paradigm. As can be seen in these tables and figures, as α grows, the performance of the frame semantic-based model has a rising trend

Table 8.8: QA runs with different α values on 75 TREC 2004 factoid questions - FSFE-NFS method in the frame semantic-based answer processing model, bold numbers show maximum values in each column

α value	*mrr*					
	strict			lenient		
	FSB	ENB	fused	FSB	ENB	fused
0.00	0.027	**0.400**	0.427	0.040	**0.413**	0.453
0.05	0.040	0.400	0.440	0.053	**0.413**	0.467
0.10	0.040	0.400	0.440	0.053	**0.413**	0.467
0.15	0.080	0.360	0.440	0.093	0.373	0.467
0.20	0.133	0.293	0.427	0.160	0.307	0.467
0.25	0.200	0.240	0.440	0.227	0.253	0.480
0.30	0.240	0.213	0.453	0.280	0.227	0.507
0.35	0.293	0.200	0.493	0.347	0.213	0.560
0.40	0.360	0.147	0.507	0.413	0.160	0.573
0.45	0.360	0.147	0.507	0.413	0.160	0.573
0.50	0.413	0.107	0.520	0.467	0.120	0.587
0.55	0.467	0.067	0.533	0.520	0.080	0.600
0.60	0.493	0.053	0.547	0.547	0.067	0.613
0.65	0.533	0.040	0.573	0.587	0.053	**0.640**
0.70	0.560	0.027	**0.587**	0.613	0.040	0.653
0.75	0.573	0.000	0.573	0.627	0.000	0.627
0.80	0.573	0.000	0.573	0.627	0.000	0.627
0.85	**0.587**	0.000	**0.587**	**0.640**	0.000	**0.640**
0.90	**0.587**	0.000	**0.587**	**0.640**	0.000	**0.640**
0.95	**0.587**	0.000	**0.587**	**0.640**	0.000	**0.640**
1.00	**0.587**	0.000	**0.587**	**0.640**	0.000	**0.640**

causing the *mrr* values of the entity-base model to drop in a quite dramatic way. With the convex linear function shown in Equation 8.3, these trends are expected since the bias towards the answer candidates extracted by the frame semantic-based model is increased and these answer candidates get higher scores than those of the answer candidates extracted by the entity-based model. However, an important observation can be made on the points at which $\alpha = 0.00$ and $\alpha = 1.00$. For the former, the overall *mrr* is supposed to be equal to the *mrr* of the entity-based model. In contrast, there is a minor difference as the *mrr* value of the frame semantic-based model is greater than 0.0. This phenomenon is due to the fact that the entity-based model fails to report any answers for a few questions and returns nil sets; therefore, the counterpart frame semantic-based model reports its answers which are scored 0.0 (as a result of $\alpha = 0.00$) as the only possibilities. Not surprisingly, some of these 0.0-scored answers are correct. At $\alpha = 1.00$ a similar phenomenon happens where the *mrr* of the entity-based model is expected to be 0.0 and the overall *mrr* to be equal to the *mrr* of the frame semantic-based model. The expectation is satisfied as there are no 0.0-scored correct answers obtained by the entity-based model.

Another aspect of these results is the peaks of the overall curves in Figure 8.1 and Figure 8.2 which are achieved at large α values ($\alpha = 0.70$ and $\alpha \geq 0.85$ in Figure 8.1 and $\alpha \geq 0.60$ in Figure 8.2) where the emphasis of the answer processing task is on the frame semantic-based model. This shows

Table 8.9: QA runs with different α values on 75 TREC 2004 factoid questions - FSFE-FS method in the frame semantic-based answer processing model, bold numbers show maximum values in each column

α value	*mrr*					
	strict			lenient		
	FSB	ENB	fused	FSB	ENB	fused
0.00	0.027	**0.400**	0.427	0.027	**0.413**	0.440
0.05	0.147	0.253	0.400	0.173	0.267	0.440
0.10	0.333	0.173	0.507	0.387	0.187	0.573
0.15	0.400	0.093	0.533	0.493	0.107	0.600
0.20	0.520	0.027	0.547	0.573	0.027	0.600
0.25	0.560	0.013	0.573	0.613	0.013	0.627
0.30	0.573	0.013	0.587	0.627	0.013	0.640
0.35	0.573	0.013	0.587	0.627	0.013	0.640
0.40	0.587	0.013	0.600	0.640	0.013	0.653
0.45	0.600	0.000	0.600	0.653	0.000	0.653
0.50	0.600	0.000	0.600	0.653	0.000	0.653
0.55	0.600	0.000	0.600	0.653	0.000	0.653
0.60	**0.627**	0.000	**0.627**	**0.680**	0.000	**0.680**
0.65	**0.627**	0.000	**0.627**	**0.680**	0.000	**0.680**
0.70	**0.627**	0.000	**0.627**	**0.680**	0.000	**0.680**
0.75	**0.627**	0.000	**0.627**	**0.680**	0.000	**0.680**
0.80	**0.627**	0.000	**0.627**	**0.680**	0.000	**0.680**
0.85	**0.627**	0.000	**0.627**	**0.680**	0.000	**0.680**
0.90	**0.627**	0.000	**0.627**	**0.680**	0.000	**0.680**
0.95	**0.627**	0.000	**0.627**	**0.680**	0.000	**0.680**
1.00	**0.627**	0.000	**0.627**	**0.680**	0.000	**0.680**

that in a joint answer processing procedure where the entity-based model is not very sophisticated, it is wise to put emphasis on a frame semantic-based model which can identify answer candidates from a comprehensively annotated text environment.

The fact that the peak of the fused *mrr* values is not reached at $\alpha = 0.50$ is remarkable. This indicates the advantage of conducting a tuning/learning procedure before using the score-based fusion technique through finding an optimal value for the convex linear parameter α. With such a value for α, the convex combination used in our work introduces a better approach than the simple equally weighting approaches to achieve the maximum possible coverage over the correct answers.

By considering the results obtained with the FSFE-NFS and FSFE-FS methods in the frame semantic-based answer processing models, it can be seen that the stronger answer scoring technique used in the FSFE-FS method makes the dominance of the frame semantic-based model more visible. The *mrr* of the entity-based model drops[3] more drastically in Figure 8.2 compared to that in Figure 8.1 where the FSFE-NFS is used. This results in a higher overall answer processing performance when using FSFE-FS (0.627 in Table 8.9) instead of FSFE-NFS (0.587 in Table 8.8).

The fact that the maximum fused *mrr* values can be achieved at $\alpha = 1.00$ raises this question

[3] The drop in the ENB performance happens as the ENB answers are demoted (lowly scored) as a result of applying the convex fusion function shown in Equation 8.3.

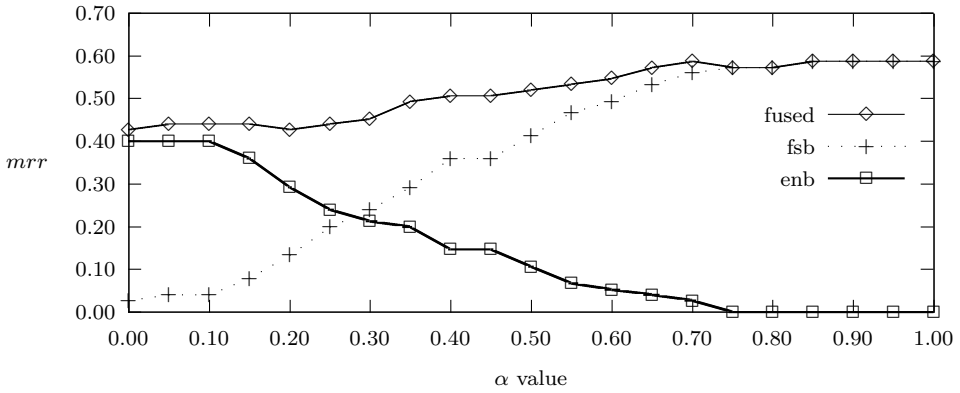

Figure 8.1: The trends of strict answer processing *mrr*s for the two answer processing models and fused answer processing performance - FSFE-NFS method in the frame semantic-based model

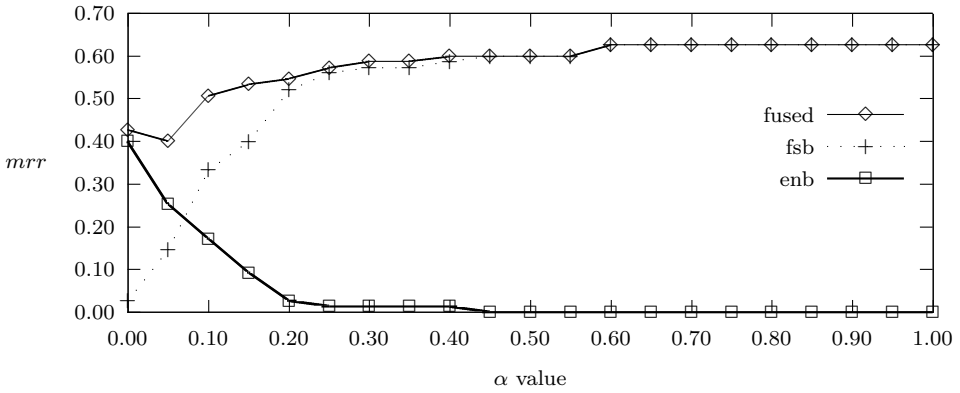

Figure 8.2: The trends of strict answer processing *mrr*s for the two answer processing models and fused answer processing performance - FSFE-FS method in the frame semantic-based model

"Why the entity-based model should be used at all?". To answer this question we conduct another study in the following subsection.

8.3.2 Correct Answer Coverage

In order to achieve the maximum possible answer processing performance using two answer processing models - the entity-based and frame semantic-based models - another important aspect is the analysis of their individual and combined correct answer coverage.

As shown in Figure 8.3, there are four possible situations in answer coverage by these two answer processing models. The best case that can be achieved is Case 2 where the two models cover different sets of correct answers. Case 3 and Case 4 are not desirable as the results in one of the

models are totally redundant. Case 1, however, is the most probable situation because of the different characteristics and capabilities of the two models. This means that there are always questions that can only be answered by a single answer processing module using its specific approach as well as the questions which are answerable using both models.

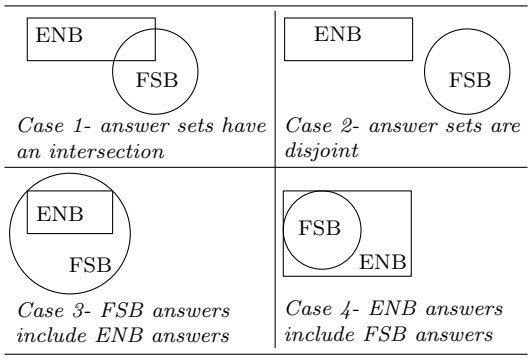

Figure 8.3: Correct answer coverage schemes by two answer processing models

According to our analysis on the answer sets of the entity-based and frame semantic-based models in the TREC 2004 factoid questions, Case 1 answer coverage scheme holds. For this analysis, we have conducted two QA runs to extract factoid answers using two answer processing settings, FSB-only and ENB-only. Table 8.10 shows the *mrr* values for the two models separately and combined with each other (score-based fusion with $\alpha = 0.50$) using two methods of frame semantic-based answer processing, FSFE-NFS and FSFE-FS. The frame semantic-based model extracts answers at the highest annotation level SHAL-HL. The SHALMANESER parser used for the frame semantic-based model is trained with FrameNet 1.2 and the human corrections are performed on the automated annotations using the frameset in FrameNet 1.2.

Table 8.10: *mrr* values for the individual answer processing models and their combinations using score-based fusion with $\alpha = 0.50$ on 75 TREC 2004 factoid questions

QA method	*mrr*	
	strict	lenient
1) ENB-only	0.400	0.413
2) FSB-only: FSFE-NFS	0.587	0.640
3) FSB-only: FSFE-FS	**0.627**	**0.680**
1 and 2	0.520	0.587
1 and 3	0.600	0.653

There are different possible situations that can happen in Case 1 which are shown in Figure 8.4. In real processes, Coverage 4 is the most probable case. This is because attribution of answers to the answer processing models is affected by the fusion mechanism. A strong bias towards either model

can detract the number of correct answers attributed to the other model. This occurs as there are answers which can be identified by both models and based on the settings of the fusion method, answers of either model may get the priority and be reported as the final answer. Coverage 2 and 3 are possible if one of the models, under specific biases or conditions, can strongly dominate.

Coverage 1 is the result that would ideally be achieved. We perform an analysis on the answer sets of the two answer processing models to examine the difference between the ideal performance and actual performances. The ideal performance could be reached by extracting the union set of the correct answer sets of the two individual models (Coverage 1 or possible upper bound). We measure the Coverage 1 *mrr* values by manually compiling the answer files of the two answer processing models. We take all correct answers from one model and add all correct answers from the second model which have not been extracted by the first model.

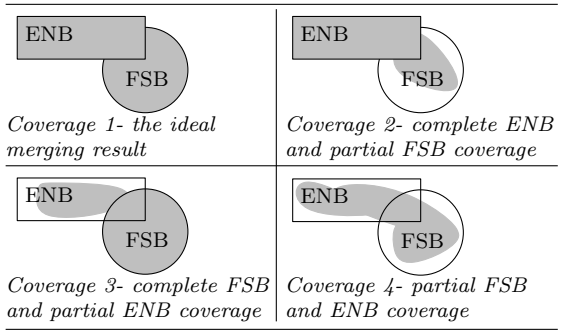

Figure 8.4: Distribution of correct answers in Case 1 of Figure 8.3

The actual performances are those achieved in our previous experiments by using different α values. We select the points where the two models are equally weighted and where the highest *mrr* value is achieved. Table 8.11 summarizes the results of this analysis.

Table 8.11: *mrr* values on 75 TREC 2004 factoid questions at important answer coverage points

QA method	Fused *mrr*			
	ENB+FSFE-NFS		ENB+FSFE-FS	
	strict	lenient	strict	lenient
Equally weighted ($\alpha = 0.50$)	0.520	0.587	0.600	0.653
Best merged	0.587	0.640	0.627	0.680
Possible upper bound (= Coverage 1)	**0.667**	**0.733**	**0.706**	**0.773**

After our analysis on the correct answer sets using both methods of frame semantic-based answer processing (FSFE-NFS and FSFE-FS), coverage of the correct answers varies from Coverage 2 to Coverage 4. Coverage 1 could show an increase from the value of the best merged to the value of the

possible upper bound (0.587 to 0.667). This remains a possible goal to be achieved. This improvement would require different techniques of answer processing and the exploitation of the entity-based model in our current and future experiments in conjunction with the frame semantic-based model. In order to achieve the possible upper bound of answer coverage, employing a comprehensive fusion technique with optimal settings is required.

8.4 Summary

In this chapter, two methods for fusing answer lists extracted by two different answer processing models - the entity-based and frame semantic-based models - are studied. The first method - rank-based fusion - uses answer ranks and their redundancies in merging answer lists, sorting the answers, and selecting a final top-ranked answer. The second method - score-based fusion - uses answer scores as well as their redundancies. These answer scores are obtained from the calculations performed in each answer processing model.

By conducting different factoid answer processing experiments, it has been shown that the score-based fusion method performs slightly better than the rank-based technique giving more opportunity for the frame semantic-based model to retrieve more correct answers. We have, therefore, carried out further studies on the score-based method. First, an analysis with respect to tuning the linear convex parameter has been performed. The conclusion was that equally weighting answer processing models is not always the best way for obtaining a maximum overall *mrr*. Second, an investigation has been conducted in order to observe the answer coverage rates that exist on the two answer sets of the two answer processing models. It has been concluded that each answer processing model extracts a set of answers that has an intersection with that of the other answer processing models. It is still a possible future goal for a QA system to retrieve an overall set of answers which covers the union set of the correct answers of each model.

Chapter 9

Conclusion

This final chapter concludes the thesis with a summary of the work carried out, the main contributions of this study, and a discussion on future work.

9.1 Recapitulation

This thesis investigated the impact of frame semantics on natural language factoid QA concentrating on two main sub-tasks of a pipelined QA architecture, namely passage retrieval and answer processing.

In the passage retrieval phase, we have considered two aspects that can be enhanced:

i) The scoring and ranking algorithm of retrieved passages, and
ii) The input query formulation strategy.

The enrichment of passage retrieval systems can be achieved by using frame semantics and syntactical information solely or in conjunction with other types of information such as keyword-based, topical, or passage length information.

In our work on passage retrieval ranking algorithms, we have modified the MultiText retrieval method to make its passage scoring and ranking function more suitable for the TREC QA task. In this thesis, however, emphasis has been placed on input query formulation to enhance the performance of passage retrieval systems for QA. We have developed a frame semantic-based boosting cycle which converges to the best input query to maximize the chance of retrieving the most specific passages to a given question. The boosting cycle is based on reformulating the query by substituting its main predicate with other LUs. These LUs are those of the same FrameNet frame from which the main predicate of the query inherits.

Studying the utility of frame semantics in factoid QA also leads to the investigation of the answer processing performance. We have identified and tested four main facets that directly affect the

answer processing performance when a QA system benefits from a frame semantic-based model solely or in conjunction with other approaches:

i) The level of semantic class identification and semantic role labeling. Four levels of frame semantic parsing have been introduced and their impact on factoid answer processing performance has been tested.

ii) The technique of semantic alignment of question and passage frames along with the answer scoring technique. Five different frame and FE alignment techniques have been developed including CFFS, FSFE-NFS, FSFE-FS, FE-NFES, and FE-FES (see Chapter 6). The CFFE and FSFE-NFS methods have been used in previous works, while the other techniques are new approaches.

iii) The FrameNet lexical coverage. The effect of FrameNet coverage on factoid answer processing performance has been quantified with respect to the different part-of-speech predicates in the two FrameNet datasets 1.2 and 1.3.

iv) The fusion method for fusing the results of the frame semantic-based answer processing model with those of other models. Two fusion methods for merging the answer lists of the frame semantic-based model and an entity-based model have been proposed: a) score-based fusion, and b) rank-based fusion. We have focused on the score-based method because of its superior performance compared to that of the rank-based method. We have further evaluated the convex linear function used in this method for weighting answers. This leads to retrieving the maximum number of correct answers by the two individual models and distinguishes our work from the existing approaches that equally weight the answers from different answer processing models.

9.2 Contributions

The main contributions of this thesis have been made regarding two sub-tasks of QA:

- In enhancing answer passage retrieval effectiveness for factoid QA systems, we have found that:

  i) Using the main topic of a given question, the limited syntactic information of the part-of-speech of the terms, the density-based information about the terms, the information on the length of retrieved passages, and the rate of coverage of the passages over the query terms in the process of scoring and ranking retrieved passages improves the effectiveness of answer passage retrieval. We have developed a modified MultiText passage retrieval method which uses these types of information and significantly improves the effectiveness of the baseline MultiText algorithm and most of the Lemur passage retrieval methods.

  ii) The scenario-based relations between the LUs in FrameNet frames, used in frame semantic-based boosting, can have a positive impact on the surface mismatch resolution between a given question and the text of related documents. This results in retrieving answer

passages for a greater number of questions or improving the rank of the answer passages. Improvements achieved by frame semantic-based boosting, however, are not statistically significant at this stage. With higher FrameNet coverage over predicates, it is expected to achieve more improvement over the methods which are not semantically boosted. This is because with higher lexical coverage there will be more opportunities for the semantic boosting cycle to converge to the best query.

- Our contributions in the analysis of FrameNet-based answer processing include the following:
  i) In analysing the effect of different levels of shallow semantic parsing on the answer processing performance we have found that:
     - A poorly performing shallow semantic parser which generates sparse annotations with low accuracy cannot assist the answer processing task.
     - The performance of a frame semantic-based answer processing model is solidly enhanced when the semantic role labeling task is augmented with manual FE assignments compared to corrected semantic class identification.
     - There is a need for more work on encapsulating non-verb predicates information in FrameNet. Our study has shown that the performance of a frame semantic-based answer processing model increases when FEs of both verb and non-verb frames are manually corrected.
  ii) The analysis of five FrameNet-based answer processing techniques proposed and developed in this thesis has shown that:
     - The complete frame and FE alignment technique (as in CFFE) is not a suitable technique for answer processing and does not achieve high answer rankings as measured by mrr. This is because of the presence of textual string mismatches between FE instances and also due to the existence of predicate chains in answer passages in some cases. A specific FE alignment method (such as that in FSFE-NFS) can better identify answer candidates.
     - Our new frame scoring mechanism in the FSFE-FS method, based on the question context in frame-evoking sentences, in a partial FE alignment procedure has led to the best answer processing performance among the five answer processing methods.
     - Relaxation of the frame and FE alignment task to just FE alignment (as in FE-NFES and FE-FES), without taking into consideration the FE-containing frames, results in poorer answer processing performance.
     - With inaccurate and incomplete sparse annotations, the different techniques of semantic alignment do not promise any significant difference in answer processing performances.
  iii) By testing the effect of FrameNet lexical coverage on the performance of FrameNet-based factoid answer processing, we have found that:

- As expected, the higher FrameNet coverage over predicates results in higher factoid answer processing effectiveness in an accurate and complete annotation environment. However, the improvements achieved by a higher lexical coverage (in FrameNet 1.3 compared to FrameNet 1.2) are not statistically significant at this time.
- With the current growth ratio of different part-of-speech predicates in FrameNet datasets, noun predicates are in a crucial situation. Therefore, the work on covering more nouns in FrameNet is important in enhancing the factoid answer processing performance at this stage.
- The higher FrameNet coverage in a sparse and inaccurate annotation environment may damage the performance of the answer processing model.

iv) The analysis of our two fusion techniques for answer list merging has shown that:

- Overall, the baseline entity-based answer processing model is outperformed by using score-based and rank-based fusion methods in a merged answer processing setting which fuses the results of a frame semantic-based model with those of the entity-based model.
- The score-based method generally performs better than the rank-based method in an accurate and complete annotation environment. In the absence of accurate annotations, the rank-based strategy slightly outperforms the score-based method.
- The negative effect of a poor frame semantic-based model, due to poor text annotation, on the performance of an entity-based model is higher in the score-based fusion method.
- By using an articulated fusion function in the score-based method, like the convex linear function used in this thesis, it is possible to enhance the overall retrieval of correct answers compared to the methods that equally emphasize the answers of different models.

9.3 Epilogue: Frame Semantics Helps QA

Our work has established that frame semantics can assist factoid QA systems in answering questions that are difficult to answer by existing QA approaches and other linguistic resources. This can be done by: i) retrieving a greater number of specifically related passages which actually contain answer candidates, and ii) extracting correct answers from answer passages with scenario-based semantic relatedness between questions and answer-containing sentences. However, there are still difficulties in exploiting frame semantics in different parts of a factoid QA system. Two such major challenges are: a) improving the level of shallow semantic parsing accuracy, and b) conducting a meta-learning process to characterize the problems (questions) that can be effectively solved (answered) by a frame semantic-based model. It is worthwhile to tackle these bottlenecks to improve the upper

bound factoid QA performance that can be achieved by such a meaning-aware approach.

9.4 Future Directions

In enhancing passage scoring and ranking functions, a possible direction for future work is to use FrameNet-based information in passages in addition to the information that we have used in this thesis. FrameNet-based information in passages has previously been used for passage scoring in (Hickl et al. 2006); however, the study of a combination of our passage scoring parameters with the frame semantic structure of passages has yet to be conducted.

In semantically boosting passage retrieval effectiveness, the decision-making algorithm for keeping or substituting a query term with an alternative LU in the question frame is subject to further analysis. In this thesis, we have considered the information retrieval-based logic to interpret the changes of the maximum and minimum passage scores. It is still possible to learn more sophisticated pieces of reasoning so that the boosting algorithm converges to the best queries in a more effective and efficient way.

The improvements in precision and recall measures in some of our passage retrieval experiments are statistically significant. In terms of mrr, however, some improvements are at higher p levels. From an information retrieval point of view this is promising; nevertheless, in the context of QA it means that the enhancement of the mrr measures is still possible and could further assist the answer processing phase with higher ranked answer candidates.

To more precisely study the effect of the semantic boosting cycle (described in this thesis) for improving answer passage retrieval performance, a follow-up direction is to consider the Detour system (Burchardt, Erk, and Frank 2005) in order to identify the best match frames that describe the question predicates. This can overcome the current incomplete coverage of FrameNet, although it is not yet guaranteed that the best match frame can suggest more specific query terms to enhance the ranking of potential answer-containing passages.

The semantic coverage can be further elaborated by widening the scope of the boosting cycle to a broader semantic domain which includes inter-related frames in FrameNet. By taking this approach, the boosting cycle will be more time-consuming depending on the number of levels of relations to be explored. This will necessitate the emergence of a more efficient convergence function.

To further work on boosting retrieval effectiveness through query rewriting, we intend to compare our boosting cycle with those proposed in (Moldovan et al. 1999) and (Harabagiu et al. 2000). This will require the development of a framework configuration which includes different boosting methodologies over the same dataset. It will then be possible to specifically show which boosting procedure is more effective for answer passage retrieval.

In order to more precisely distinguish between the improvements that frame evocation and FE

assignment tasks can contribute to answer processing, it is necessary to set an intermediate annotation level between SHAL-AF and SHAL-HL with human level frame evocation which includes synthesized noises in the FE assignment task. This level of annotation will have human level frame evocation and synthesized machine level FE assignment. Generating the synthesized noise over the FE assignment task, however, requires a thorough analysis of the factors which interfere with a human level complete FE assignment procedure in automated shallow semantic parsers. From this viewpoint, the noise may include different types of miss-assignments such as wrong semantic role labeling, incomplete string allocations to the FEs, not assigning a FE to an existing sentence segment that is playing the semantic role of the FE, and so forth.

It is interesting to investigate the impact of shallow semantic parsing levels on other related applications. For instance, Information Extraction and Semantic Extraction using frame semantics as in (Mohit and Narayanan 2003; Moschitti, Morarescu, and Harabagiu 2003), Machine Translation using frame semantics as studied in (Boas 2002; Fung and Chen 2004; Sachs 2004), and Semantic Textual Entailment with FrameNet frames and FEs, as considered in (Burchardt 2006; Burchardt and Frank 2006; Burchardt et al. 2007), are areas of interest where the impact of shallow semantic parsing levels can be evaluated.

The effect of FrameNet coverage on semantic alignment-based answer processing can be somewhat alleviated by employing assisting tools such as the Detour system to identify best match frames in cases where there is no exact frame evoked by a predicate. Because of time limitations, we did not use Detour in our gold standard annotation. It would have required a revision of the whole annotation task to maintain the consistency of the methodology. With the impact of FrameNet coverage on the answer processing performance demonstrated in this thesis, using Detour promises an improvement over the factoid answer processing performance.

With the current limitations of frame semantic-based answer processing, the joint application of this model with other existing models requires a more sophisticated analysis. There is still a gap in reaching the maximum point of correct answer retrieval which includes the union of all correct answers of the two answer processing models. This may require an offline analysis to weight the models in accordance with parameters such as the answer type, questions stems, and syntactical structure of questions.

We would also like to observe the improvement of the overall answering *mrr* of a more sophisticated non-frame-semantic-based answer processing module, such as LCC's entity-based model that employs CICERO LITE NE tagger[1], when joined with the frame semantic-based answer processing models that we have proposed in this thesis.

Finally, we have conducted some experiments in terms of *fusing* the frame semantic-based QA model with other models; however, we have not studied the *combination* of such models. Characterization of the conditions necessary for applying a frame semantic-based model in answer processing

[1] http://www.languagecomputer.com/solutions/information_extraction/cicero_lite/index.html

requires another investigation. This can include analysis of a comprehensive learning process which involves a feature analysis (feature extraction) procedure to characterize problems (questions) that can be more effectively handled using frame semantic-based QA models.

Appendix A

Extra Tables and Figures

Table A.1: Parameter set for document indexing using Lemur for the MultiText passage retrieval algorithm

Parameter	Definition	Value
Index	Full path to index file containing the name of index	...\target$i$\doc_index\trec06_aquaint_doc_index-target$i$ (i denotes the TREC target number)
Index type	The type of index	inv (for inverted file)
Memory	Memory in bytes for buffering purposes	128000000
Position	To keep term positions in documents or not	1
Stemmer	The stemmer to stem the terms	Porter
Count stop-words	Whether to count stop-words or not	true
Document format	Reveals the format of the documents	TREC
Data files	The name of the file containing the files to be indexed	...\target$i$\doc_url\doc_url_list.i (i denotes the TREC target number)

Table A.2: Parameter set for passage indexing using Lemur

Parameter	Definition	Value
Index	Full path to index file containing the name of index	...\target$i$\pass_index\trec06_aquaint_pass_index-target$i$ (i denotes the TREC target number)
Index type	The type of index	inv (for inverted file)
Memory	Memory in bytes for buffering purposes	128000000
Position	To keep term positions in documents or not	1
Stemmer	The stemmer to stem the terms	Porter
Count stop-words	Whether to count stop-words or not	true
Document format	Reveals the format of the documents	TREC
Data files	The name of the file containing the files to be indexed	...\ApplicationFiles\PassageIndexer_list.txt
Passage size	The fixed length of passages to be indexed	300

Table A.3: Parameter set for passage retrieval using Lemur

Parameter	Definition	Value
Retrieval model	The model to be used in passage (/document) retrieval	0: TF/IDF 1: OkapiBM25 2: KL_DivergenceLangauge 3: InQuery_CORI 4: CORI_collection_selection 5: Cosine
Index	Full path to index file containing the name of index	...\target$i$\pass_index\trec06_aquaint_pass_index-target$i$.ifp (i denotes the TREC target number)
TREC result format	The format of the result records in the result file	0 (for non-TREC format with simple three columns)
Result count	The number of passages (/documents) to retrieve	20
Result file	The result file	PassageRetrievalResult.txt
Text query	The file containing the query stream	PassageRetrievalQuery.txt

By assuming that:

- The population proportion is equal to the sample proportion, and
- The sampling task is random

we can consider the formula below for estimating the population proportion:

$$n \geq \frac{{z_c}^2 \times (p \times q)}{e^2} \tag{1}$$

where:

n is the minimum required sample size,
e is the margin of error,
z_c is the z-score obtained from a normal table,
p is the sample proportion, and
$q = p - 1$

Therefore, with:

- The confidence level = 95%, and
- The margin of error $(e) = \pm 0.03$

Equation 1 can be rewritten as:

$$n \geq \frac{1.96^2 \times (p \times q)}{0.03^2} \tag{2}$$

where:

$$max(p \times q) = 0.25 \tag{3}$$

From Equation 2 and Equation 3, we have:

$$n = ceil(\frac{1.96^2 \times (0.25)}{0.03^2}) = 1068 \tag{4}$$

Figure A.1: Minimum number of samples required for estimating population proportion at the confidence level 95% and precision ± 0.03

Appendix B

Acronyms

acc	Accuracy
AF	All Frames
AP	Answer Processing
APd	All Predicates
Avg.	Average
BL	Baseline
CFFE	Complete Frame and FE alignment - No Frame Scoring
EAT	Expected Answer Type
ENB	Entity-Based
FE	Frame Element
FE-NFES	FE alignment - No FE Scoring
FE-FES	FE alignment - FEs Scored
FEE	Frame Evoking Element
FN	FrameNet
FN-FEE	FrameNet FEEs
FRED	FRame Disambiguator
FSB	Frame Semantic-Based
FSFE-NFS	Frame alignment with Specific FE matching - No Frame Scoring
FSFE-FS	Frame alignment with Specific FE matching - Frames Scored
HL	Human Level
IDF	Inverse Document Frequency
IR	Information Retrieval
ln	lenient
LU	Lexical Unit

Max-FEEA	Maximum of the FEEs labelled by Annotators
mrr	Mean Reciprocal Rank
NE	Named Entity
POS	Part-Of-Speech
prec	Precision
QA	Question Answering
rec	Recall
ROSY	ROle assignment SYstem
SALTO	the SALsa annotation TOol
SHAL	SHALMANESER
SHALMANESER	a SHALlow seMANtic parSER
st	strict
TE	Ternary Expression
TF	Term Frequency
TREC	Text REtrieval Conference
Un-FEEA	Union of the FEEs labelled by Annotators
VF	Verb Frames

Bibliography

Amoia, Marilisa and Claire Gardent (2005). "Recognition of alternation paraphrases: A robust and exhaustive symbolic approach." In: *Knowledge and Reasoning for Answering Questions (KRAQ 2005)*. Edinburgh, Scotland.

Baker, Collin F., Charles J. Fillmore, and John B. Lowe (1998). "The Berkeley FrameNet project." In: *17th International Conference on Computational Linguistics (COLING 1998)*. Vol. 1. Universite de Montreal, Montreal, Quebec, Canada, pp. 86–90.

Bernardi, Raffaella, Valentin Jijkoun, Gilad Mishne, and Maarten de Rijke (2003). "Selectively using linguistic resources throughout the question answering pipeline." In: *2nd CoLogNET-ElsNET Symposium*.

Bilotti, Matthew W., Paul Ogilvie, Jamie Callan, and Eric Nyberg (2007). "Structured retrieval for question answering." In: *30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2007)*. Amsterdam, The Netherlands: ACM, pp. 351–358.

Bittner, Thomas, Maureen Donnelly, and Barry Smith (2004). "Individuals, universals, collections: On the foundational relations of ontology." In: *Third Conference on Formal Ontology in Information Systems (FOIS 2004)*. Torino, Italy: IOS Press, pp. 37–48.

Boas, Hans C. (2002). "Bilingual FrameNet dictionaries for machine translation." In: *Third International Conference on Language Resources and Evaluation*. Ed. by M. Gonzalez Rodriŋguez Araujo and C. Paz Suarez. Vol. 4. Las Palmas, Spain, pp. 1364–1371.

Bos, Johan (2006). "The "La Sapienza" question answering system at TREC 2006." In: *Fifteenth Text Retrieval Conference (TREC 2006)*.

Bosch, Peter and Bart Geurts (1989). *Proccessing definite NPs. IBM. Lilog project*. Tech. rep.

Brill, Eric, Susan Dumais, and Michele Banko (2002). "An analysis of the AskMSR question-answering system." In: *ACL-02 Conference on Empirical Methods in Natural Language Processing (EMNLP '02)*. Association for Computational Linguistics, pp. 257–264.

Brill, Eric, Jimmy Lin, Michele Banko, Susan Dumais, and Andrew Ng (2001). "Data-intensive question answering." In: *Tenth Text Retrieval Conference (TREC 2001)*, pp. 393–401.

Burchardt, Aljoscha (2006). "Approaching textual entailment with LFG and FrameNet frames." In: *PASCAL Challenges Workshop 2*. Venice, Italy.

Burchardt, Aljoscha, Katrin Erk, and Anette Frank (2005). "A WordNet detour to FrameNet." In: *2nd GermaNet Workshop*. University of Bonn, Germany.

Burchardt, Aljoscha and Anette Frank (2006). "Approximating textual entailment with LFG and FrameNet frames." In: *Second PASCAL Recognizing Textual Entailment Workshop*. Venice, Italy, pp. 92–97.

Burchardt, Aljoscha, Katrin Erk, Anette Frank, A. Kowalski, and Sebastian Pado (2006). "SALTO – A versatile multi-level annotation tool." In: *Fifth International Conference on Language Resources and Evaluation (LREC 2006)*. Genoa, Italy.

Burchardt, Aljoscha, Nils Reiter, Stefan Thater, and Anette Frank (2007). "A semantic approach to textual entailment: System evaluation and task analysis." In: *ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*. Prague, Czech Republic.

Cardie, Claire, Vincent Ng, David Pierce, and Chris Buckley (2000). "Examining the role of statistical and linguistic knowledge sources in a general-knowledge question-answering system." In: *Sixth Applied Natural Language Processing Conference (ANLP-2000)*. Seattle, Washington, USA, pp. 180–187.

Carroll, John, Ted Briscoe, and Antonio Sanfilippo (1993). "Parser evaluation : A survey and a new proposal." In: *First Conference on Linguistic Resources*. Canada, pp. 447–455.

Chai, Joyce Y. and Rong Jin (2004). "Discourse structure for context question answering." In: *HLT-NAACL 2004 Workshop on Pragmatics of Question Answering*. Ed. by Sanda Harabagiu and Finley Lacatusu. Boston, US: Association for Computational Linguistics, pp. 23–30.

Chen, John and Owen Rambow (2003). "Use of deep linguistic features for the recognition and labeling of semantic arguments." In: *Conference on Empirical Methods in Natural Language Processing (EMNLP 2003)*. Sapporo, Japan.

Choquette, Martin (1996). "Passage retrieval." MPhile course in computer speech and language processing. University of Cambridge.

Chu-Carroll, Jennifer, Krzysztof Czuba, John Prager, and Abraham Ittycheriah (2003). "In question answering, two heads are better than one." In: *2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology (NAACL '03)*. Edmonton, Canada: Association for Computational Linguistics, pp. 24–31.

Clarke, Charles L.A., Gordon V. Cormack, and F. Burkowski (1995). "Shortest substring ranking (MultiText experiments for TREC-4)." In: *Fourth Text Retrieval Conference (TREC-4)*, pp. 295–304.

Clarke, Charles L.A., Gordon V. Cormack, and Elizabeth A. Tudhope (2000). "Relevance ranking for one to three term queries." In: *Information Processing and Management* 36.2, pp. 291–311.

Clarke, Charles L.A. and Egidio L. Terra (2003). "Passage retrieval vs. document retrieval for factoid question answering." In: *26th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*. Toronto, Canada: ACM Press, pp. 427–428.

Clarke, Charles L.A., Gordon V. Cormack, D. Kisman, and T. Lynam (2000). "Question answering by passage selection (MultiText experiments for TREC-9)." In: *Ninth Text Retrieval Conference (TREC-9)*, pp. 673–684.

Clarke, Charles L.A., Gordon V. Cormack, T.R. Lynam, C.M. Li, and G.L. McLearn (2001). "Web reinforced question answering (MultiText experiments for TREC 2001)." In: *Tenth Text Retrieval Conference (TREC 2001)*, pp. 673–679.

Cohen, Jacob (1960). "A coefficient of agreement for nominal scales." In: *Educational and Psychological Measurement* 20, pp. 37–46.

Collins, Allan M. and Ryszard S. Michalski (1989). "The logic of plausible reasoning: A core theory." In: *Cognitive Science* 13.1, pp. 1–49.

Cook, Walter A. (1989). *Case grammar theory*. Georgetown University Press.

Cormack, Gordon, C. Palmer, M. Biesbrouck, and C. Clarke (1998). "Deriving very short queries for high precision and recall (MultiText experiments for TREC-7)." In: *Seventh Text Retrieval Conference (TREC-7)*, pp. 121–133.

Cortes, Corinna and Vladimir Vapnik (1995). "Support-Vector Networks." In: *Machine Learning* 20.3, pp. 273–297.

Cui, Hang, Renxu Sun, Keya Li, Min-Yen Kan, and Tat-Seng Chua (2005). "Question answering passage retrieval using dependency relations." In: *28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2005)*. Salvador, Brazil: ACM, pp. 400–407.

Dang, Hoa Trang, Jimmy Lin, and Diane Kelly (2006). "Overview of the TREC 2006 question answering track." In: *Fifteenth Text Retrieval Conference (TREC 2006)*.

Dolbey, Andrew, Michael Ellsworth, and Jan Scheffczyk (2006). "BioFrameNet: A domain-specific FrameNet extension with links to biomedical ontologies." In: *KR-MED 2006 "Biomedical Ontology in Action"*. Baltimore, Maryland, US.

Dumais, Susan, Michele Banko, Eric Brill, Jimmy Lin, and Andrew Ng (2002). "Web question answering: Is more always better?" In: *25th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*. Tampere, Finland: ACM Press, pp. 291–298.

Elworthy, David (2000). "Question answering using a large NLP system." In: *Ninth Text Retrieval Conference (TREC-9)*, pp. 355–361.

Erk, Katrin (2006). "Frame assignment as word sense disambiguation." In: *2006 IAENG International Workshop on Computer Science (IWCS 2006)*. Tilburg University, Tilburg, The Netherlands.

Erk, Katrin and Sebastian Pado (2004). "A powerful and versatile XML format for representing role-semantic annotation." In: *Third International Conference on Language Resources and Evaluation (LREC 2004)*. Lisbon, Portugal.

— (2005). "Analyzing models for semantic role assignment using confusability." In: *Conference on Human Language Technology and Empirical Methods in Natural Language Processing (HLT/EMNLP 2005)*. Vancouver, British Columbia, Canada: Association for Computational Linguistics, pp. 668–675.

— (2006). "Shalmaneser - A toolchain for shallow semantic parsing." In: *Fifth International Conference on Language Resources and Evaluation (LREC 2006)*. Genoa, Italy.

Erk, Katrin, Andrea Kowalski, Sebastian Pado, and Manfred Pinkal (2003). "Towards a resource for lexical semantics: A large German corpus with extensive semantic annotation." In: *41st Annual Meeting on Association for Computational Linguistics (ACL 2003)*. Sapporo, Japan: Association for Computational Linguistics, pp. 537–544.

Eugenio, Barbara Di and Michael Glass (2004). "The Kappa statistic: A second look." In: *Computational Linguistics* 30.1, pp. 95–101.

Ferret, Olivier, Brigitte Grau, Martine Hurault-Plantet, Gabriel Illouz, and Christian Jacquemin (2000). "QALC - The question-answering system of LIMSI-CNRS." In: *Ninth Text Retrieval Conference (TREC-9)*.

Fillmore, Charles J. (1968). "The case for case." In: *Universals in Linguistic Theory*. Ed. by Bach and Harms, pp. 1–88.

— (1976). "Frame semantics and the nature of language." In: *Annals of the New York Academy of Sciences: Conference on the Origin and Development of Language and Speech*. Vol. 280, pp. 20–32.

Fliedner, Gerhard (2004). "Deriving FrameNet representations: Towards meaning-oriented question answering." In: *9th International Conference on Applications of Natural Language to Information Systems (NLDB 2004)*. University of Salford, Manchester, UK: Springer Berlin/Heidelberg, pp. 64–75.

Frank, Anette (2004). "Generalizations over corpus-induced frame assignment rules." In: *LREC 2004 Workshop on Building Lexical Resources from Semantically Annotated Corpora*. Lissabon, Portugal, pp. 31–38.

Fung, Pascale and Benfeng Chen (2004). "BiFrameNet: Bilingual frame semantics resource construction by cross-lingual induction." In: *20th International Conference on Computational Linguistics (COLING 2004)*. University of Geneva, Switzerland.

Gildea, Daniel and Julia Hockenmaier (2003). "Identifying semantic roles using combinatory categorial grammar." In: *Conference on Empirical Methods in Natural Language Processing (EMNLP 2003)*. Sapporo, Japan.

Gildea, Daniel and Daniel Jurafsky (2002). "Automatic labeling of semantic roles." In: *Computational Linguistics* 28.3, pp. 245–288.

Gildea, Daniel and Martha Palmer (2002). "The necessity of parsing for predicate argument recognition." In: *40th Annual Meeting on Association for Computational Linguistics (ACL 2002)*. Philadelphia, Pennsylvania: Association for Computational Linguistics, pp. 239–246.

Giuglea, Ana-Maria and Alessandro Moschitti (2006). "Shallow semantic parsing based on FrameNet, VerbNet and PropBank." In: *17th European Conference on Artificial Intelligence*. Riva del Garda, Italy, pp. 563–567.

Harabagiu, Sanda and Cosmin Adrian Bejan (2006). "An answer bank for temporal inference." In: *5th International Conference on Language Resources and Evaluation (LREC 2006)*. Genoa, Italy.

Harabagiu, Sanda and S.J. Maiorano (1999). "Finding answers in large collections of texts: Paragraph indexing + abductive inference." In: *AAAI 1999*, pp. 63–71.

Harabagiu, Sanda, G.A. Miller, and Dan Moldovan (1999). "WordNet 2 - A morphologically and semantically enhanced resource." In: *ACL-SIGLEX99: Standardizing Lexical Resources*. Maryland, pp. 1–8.

Harabagiu, Sanda, Marius Paşca, and S. Maiorano (2000). "Experiments with open-domain textual question answering." In: *18th Conference on Computational Linguistics (COLING 2000)*. Vol. 1. Saarbrucken, Germany: Association for Computational Linguistics, pp. 292–298.

Harabagiu, Sanda, Dan Moldovan, Marius Paşca, R. Mihalcea M. Surdeanu, R. Bunescu, R. Gîrju, V. Rus, and P. Morarescu (2000). "FALCON: Boosting knowledge for answer engines." In: *Ninth Text REtrieval Conference (TREC-9)*, pp. 479–489.

Harabagiu, Sanda, Dan Moldovan, Marius Paşca, Rada Mihalcea, Mihai Surdeanu, Razvan Bunescu, Roxana Gîrju, Vasile Rus, and Paul Morarescu (2001). "The role of lexico-semantic feedback in open-domain textual question-answering." In: *39th Annual Meeting on Association for Computational Linguistics (ACL 2001)*. Toulouse, France: Association for Computational Linguistics, pp. 282–289.

Harabagiu, Sanda, Dan Moldovan, C. Clark, M. Bowden, J. Williams, and J. Bensley (2003). "Answer mining by combining extraction techniques with abductive reasoning." In: *Twelfth Text Retrieval Conference (TREC 2003)*, pp. 375–383.

Harabagiu, Sanda, Dan Moldovan, C. Clark, M. Bowden, A. Hickl, and P. Wang (2005). "Employing two question answering systems in TREC-2005." In: *Fourteenth Text Retrieval Conference (TREC 2005)*.

Hermjakob, Ulf, Abdessamad Echihabi, and Daniel Marcu (2002). "Natural language based reformulation resource and web exploitation for question answering." In: *Eleventh Text Retrieval Conference (TREC 2002)*.

Hickl, Andrew, John Williams, Jeremy Bensley, Kirk Roberts, Ying Shi, and Bryan Rink (2006). "Question answering with LCC's CHAUCER at TREC 2006." In: *Fifthteenth Text Retrieval Conference (TREC 2006)*.

Honnibal, Matthew and Tobias Hawker (2005). "Identifying FrameNet frames for verbs from a real-text corpus." In: *Australasian Language Technology Workshop 2005*. Sydney, Australia, pp. 200–206.

Hovy, Eduard, Ulf Hermjakob, and Chin-Yew Lin (2001). "The use of external knowledge in factoid QA." In: *Tenth Text Retrieval Conference (TREC 2001)*, pp. 644–653.

Hovy, Eduard, L. Gerber, U. Hermjakob, M. Junk, and C-Y Lin (2000). "Question answering in Webclopedia." In: *Ninth Text Retrieval Conference (TREC-9)*, pp. 655–665.

Humphreys, Kevin, Robert Gaizauskas, Mark Hepple, and Mark Sanderson (1999). "University of Sheffield TREC-8 Q&A system." In: *Eighth Text Retrieval Conference (TREC-8)*, pp. 707–717.

Ittycheriah, Abraham, Martin Franz, and Salim Roukos (2001). "IBM's statistical question answering system–TREC-10." In: *Tenth Text Retrieval Conference (TREC 2001)*, pp. 258–265.

Ittycheriah, Abraham, Martin Franz, W.-J. Zhu, and A. Ratnaparkhi (2000). "IBM's statistical question answering system." In: *Ninth Text Retrieval Conference (TREC-9)*, pp. 229–235.

Jacquemin, Christian (1999). "Syntagmatic and paradigmatic representations of term variation." In: *37th Annual Meeting of the Association for Computational Linguistics (ACL 1999)*. Maryland, USA, pp. 341–348.

Kaisser, Michael (2005). "QuALiM at TREC 2005: Web-question answering with FrameNet." In: *Fourteenth Text Retrieval Conference (TREC 2005)*.

Kaisser, Michael and T. Becker (2004). "Question answering by searching large corpora with linguistic." In: *Thirteenth Text Retrieval Conference (TREC 2004)*.

Kaisser, Michael, S. Scheible, and B. Webber (2006). "Experiments at the University of Edinburgh for the TREC 2006 QA track." In: *Fifteenth Text Retrieval Conference (TREC 2006)*.

Kaisser, Michael and B. Webber (2007). "Question answering based on semantic roles." In: *ACL 2007 Deep Linguistic Processing Workshop (ACL-DLP 2007)*. Prague, Czech Republic.

Kaszkiel, Marcin, Justin Zobel, and Ron Sacks-Davis (1999). "Efficient passage ranking for document databases." In: *ACM Transactions on Information Systems (TOIS)* 17.4, pp. 406–439.

Katz, Boris (1990). "Using English for indexing and retrieving." In: *Artificial intelligence at MIT expanding frontiers*. MIT Press, pp. 134–165.

— (1997). "Annotating the World Wide Web using natural language." In: *5th RIAO Conference on Computer Assisted Information Searching on the Internet (RIAO 1997)*. Montreal, Canada.

Katz, Boris and J. Lin (2003). "Selectively using relations to improve precision in question answering." In: *EACL 2003 Workshop on Natural Language Processing for Question Answering*.

Katz, Boris, G. Marton, G. Borchardt, A. Brownell, S. Felshin, D. Loreto, J. Louis-Rosenberg, et al. (2005). "External knowledge sources for question answering." In: *Fourteenth Text Retrieval Conference (TREC 2005)*.

Kawahara, Daisuke, Nobuhiro Kaji, and Sadao Kurohashi (2002). "Question and answering system based on predicate-argument matching." In: *Third NTCIR Workshop on Research in Information*

Retrieval, Automatic Text Summarization, and Question Answering. Ed. by Keizo Oyama, Emi Ishida, and Noriko Kando.

Kingsbury, Paul and M. Palmer (2003). "PropBank: The next level of TreeBank." In: *Second Workshop on Treebanks and Linguistic Theories.* Ed. by J. Nivre and E. Hinrichs. Vaxjo, Sweden: Vaxjo University Press, pp. 105–116.

Kingsbury, Paul, Martha Palmer, and Mitch Marcus (2002). "Adding semantic annotation to the Penn TreeBank." In: *Human Language Technology Conference (HLT 2002).* San Diego, California.

Lee, Gary Geunbae, Jungyun Seo, Seungwoo Lee, Hanmin Jung, Bong-Hyun Cho, Changki Lee, Byung-Kwan Kwak, et al. (2001). "SiteQ: Engineering high performance QA system using lexico-semantic pattern matching and shallow NLP." In: *Tenth Text Retrieval Conference (TREC 2001)*, pp. 442–451.

Lee, Young-Shin, Young-Sook Hwang, and Hae-Chang Rim (2002). "Variable length passage retrieval for Q&A system." In: *14th Hangul and Korean Information Processing*, pp. 259–266.

Levin, Beth (1993). *English verb classes and alternations: A preliminary investigation.* Chicago: University of Chicago Press.

Light, Marc, Gideon S. Mann, Ellen Riloff, and Eric Breck (2001). "Analyses for elucidating current question answering technology." In: *Journal of Natural Language Engineering, Special Issue on Question Answering* 7.4, pp. 325–342.

Lin, Jimmy and Dina Demner-Fushman (2006). "The role of knowledge in conceptual retrieval: A study in the domain of clinical medicine." In: *29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.* Seattle, Washington, USA: ACM Press, pp. 99–106.

Lita, Lucian Vlad and Jaime Carbonell (2004). "Instance-based question answering: A data-driven approach." In: *Conference on Empirical Methods in Natural Language Processing (EMNLP 2004)*, pp. 396–403.

Litkowski, Ken (2004). "Senseval-3 task: Automatic labeling of semantic roles." In: *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text.* Barcelona, Spain, pp. 9–12.

Llopis, Fernando and Jose Vicedo (2001). "IR-n: A passage retrieval system at CLEF-2001." In: *Evaluation of Cross-Language Information Retrieval Systems: Second Workshop of the Cross-Language Evaluation Forum (CLEF 2001).* Vol. 2406. Darmstadt, Germany: Springer Berlin / Heidelberg, pp. 1211–1231.

Lonneker-Rodman, Birte (2007). *Multilinguality and FrameNet.* Tech. rep.

Lowe, John B., Collin F. Baker, and Charles J. Fillmore (1997). "A frame-semantic approach to semantic annotation." In: *SIGLEX Workshop on Tagging Text with Lexical Semantics: Why, What, and How?*

Macleod, Catherine, Ralhp Grishman, Adam Meyers, and Leslie Barrett (1998). "NOMLEX: A lexicon of nominalizations." In: *EURALEX'98*.

Marcus, Mitchell P., Beatrice Santorini, and Mary Ann Marcinkiewics (1993). "Building a large annotated corpus of English: The Penn Treebank." In: *Computational Linguistics* 19.2, pp. 313–330.

Marcus, Mitchell P., G. Kim, Mary Ann Marcinkiewics, Robert MacIntyre, Ann Bies, Mark Ferguson, Karen Katz, and Britta Schasberger (1994). "The Penn Treebank: Annotating predicate argument structure." In: *1994 Human Language Technology Workshop*. Plainsboro, New Jerey, USA: Morgan Kaufmann, pp. 110–115.

Marton, Gregory and Boris Katz (2006). "Using semantic overlap scoring in answering TREC relationship questions." In: *5th International Conference on Language Resources and Evaluation (LREC 2006)*. Genoa, Italy.

Mengel, Andreas and Wolfgang Lezius (2000). "An XML-based representation format for syntactically annotated corpora." In: *Language Resources and Evaluation Conference (LREC 2000)*. Athens, Greece.

Meyers, Adam, Catherine Macleod, Roman Yangarber, Ralph Grishman, Leslie Barrett, and Ruth Reeves (1998). "Using NOMLEX to produce nominalization patterns for information extraction." In: *Coling-ACL98 Workshop: The Computational Treatment of Nominals*. Montreal, Canada.

Meyers, Adam, Ruth Reeves, Catherine Macleod, Rachel Szekeley, Veronkia Zielinska, and Brian Young (2004*a*). "The cross-breeding of dictionaries." In: *Fourth International Conference on Language Resources and Evaluation (LREC 2004)*. Lisbon, Portugal.

Meyers, Adam, Ruth Reeves, Catherine Macleod, Rachel Szekely, Veronika Zielinska, Brian Young, and Ralph Grishman (2004*b*). "The NomBank project: An interim report." In: *HLT-NAACL 2004 Workshop: Frontiers in Corpus Annotation*. Boston, US, pp. 24–31.

Miller, George A., R. Beckwith, C. Fellbaum, D. Gross, and K.J. Miller (1990). "Introduction to WordNet: An on-line lexical database." In: *International Journal of Lexicography* 3.4, pp. 235–244.

Minsky, Marvin (1974). "A framework for representing knowledge." In: *The Psychology of Computer Vision*, pp. 211–277.

Mittendorf, E. and P. Schauble (1994). "Document and passage retrieval based on hidden Markov models." In: *Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, pp. 318–327.

Mohit, Behrang and Srini Narayanan (2003). "Semantic extraction with wide-coverage lexical resources." In: *Human Language Technology Conference (HLT-NAACL 2003)*. Edmonton, Canada, pp. 64–66.

Moldovan, Dan, M. Bowden, and M. Tatu (2006). "A temporally-enhanced PowerAnswer in TREC 2006." In: *Fifteenth Text Retrieval Conference (TREC 2006)*.

Moldovan, Dan and Vasile Rus (2001). "Logic form transformation of WordNet and its applicability to question answering." In: *39th Annual Meeting on Association for Computational Linguistics (ACL 2001)*. Toulouse, France: Association for Computational Linguistics, pp. 402–409.

Moldovan, Dan, Sanda Harabagiu, Marius Paşca, Rada Mihalcea, Richard Goodrum, Roxana Gîrju, and Vasile Rus (1999). "LASSO: A tool for surfing the answer net." In: *Eighth Text Retrieval Conference (TREC-8)*, pp. 175–184.

Moldovan, Dan, Sanda Harabagiu, Marius Paşca, Rada Miahlcea, Roxana Gîrju, Richard Goodrum, and Vasile Rus (2000). "The structure and performance of an open-domain question answering system." In: *Conference of the Association for Computational Linguistics (ACL 2000)*. Hong Kong: Association for Computational Linguistics, pp. 563–570.

Moldovan, Dan, Sanda Harabagiu, R. Gîrju, P. Morarescu, F. Lacatusu, A. Novischi, A. Badulescu, and O. Bolohan (2002). "LCC tools for question answering." In: *Eleventh Text Retrieval Conference (TREC 2002)*.

Moldovan, Dan, Christine Clark, Sanda Harabagiu, and Steve Maiorano (2003*a*). "COGEX: A logic prover for question answering." In: *2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology (NAACL-HLT 2003)*. Edmonton, Canada, pp. 87–93.

Moldovan, Dan, Marius Paşca, Sanda Harabagiu, and M. Surdeanu (2003*b*). "Performance issue and error analysis in an open-domain question answering system." In: *ACM Transactions on Information Systems (TOIS)* 21.2, pp. 113–154.

Molla, Diego (2001). "Ontologically promiscuous flat logical forms for NLP." In: *IWCS-4*. Ed. by Harry Bunt, Ielka van der Sluis, and Elias Thijsse. Tilburg University, pp. 249–265.

— (2003). "AnswerFinder in TREC 2003." In: *Twelfth Text Retrieval Conference (TREC 2003)*, pp. 392–399.

Molla, Diego and M. Gardiner (2004). "AnswerFinder at TREC 2004." In: *Thirteenth Text Retrieval Conference (TREC 2004)*.

— (2005). "AnswerFinder at TREC 2005." In: *Fourteenth Text Retrieval Conference (TREC 2005)*.

Molla, Diego and M. van Zaanen (2005). "Learning of graph rules for question answering." In: *Australasian Language Technology Workshop 2005*. Sydney, Australia, pp. 15–23.

Molla, Diego, M. van Zaanen, and L. Pizzato (2006). "AnswerFinder at TREC 2006." In: *Fifteenth Text Retrieval Conference (TREC 2006)*.

Molla, Diego, Gerold Scheinder, Rolf Schwitter, and Michael Hess (2000). "Answer extraction using a dependency grammer in ExtrAns." In: *Traitment Automatique de Lnagues (T.A.L.), Special Issue on Dependency Grammar* 41.1, pp. 127–156.

Molla, Diego, Rolf Schwitter, Fabio Rinaldi, James Dowdall, and Michael Hess (2003). "ExtrAns: Extracting answers from technical texts." In: *IEEE Intelligent Systems* 18.4, pp. 12–17.

Monz, Christof and Maarten de Rijke (2001). "Tequesta: The University of Amsterdam's textual question answering system." In: *Tenth Text Retrieval Conference (TREC 2001)*, pp. 519–529.

Moreda, Paloma, Borja Navarro, and Manuel Palomar (2005). "Using semantic roles in information retrieval systems." In: *NLDB 2005*. Alicante, Spain, pp. 192–202.

Morris, Jane and Graeme Hirst (1991). "Lexical cohesion computed by thesaural relations as an indicator of the structure of text." In: *Computational Linguistics* 17.1, pp. 21–48.

Moschitti, Alessandro, Paul Morarescu, and Sanda M. Harabagiu (2003). "Open domain information extraction via automatic semantic labeling." In: *16th International FLAIRS Conference (FLAIRS 2003)*. St. Augustine, Florida, pp. 397–401.

Moschitti, Alessandro, Bonaventura Coppola, Daniele Pighin, and Roberto Basili (2005). "Engineering of syntactic features for shallow semantic parsing." In: *ACL 2005 Workshop: Feature Engineering for Machine Learning in Natural Language Processing*. University of Michigan, Ann Arbor, Michigan, US.

Na, Seung-Hoon, In-Su Kang, Sang-Yool Lee, and Jong-Hyeok Lee (2002). "Question answering approach using a WordNet-based answer type taxonomy." In: *Eleventh Text Retrieval Conference (TREC 2002)*.

Narayanan, Srini and Sanda Harabagiu (2004*a*). "Answering questions using advanced semantics and probabilistic inference." In: *Workshop on Pragmatics of Question-Answering at HTL/NAACL 2004*. Boston, US: Association for Computational Linguistics, pp. 10–16.

— (2004*b*). "Question answering based on semantic structures." In: *20th International Conference on Computational Linguistics (COLING 2004)*. Geneva, Switzerland: Association for Computational Linguistics, pp. 693–701.

Neumann, Gunter and Bogdan Sacaleanu (2004). "Experiments on robust NL question interpretation and multi-layered document annotation for a cross-language question/answering system." In: *Workshop of the Cross-Lingual Evaluation Forum (CLEF 2004)*. Bath, UK.

Nielsen, Rodney D. and Sameer Pradhan (2004). "Mixing weak learners in semantic parsing." In: *Conference on Empirical Methods in Natural Language Processing (EMNLP 2004)*. Barcelona, Spain.

Niles, Ian and Adam Pease (2001). "Towards a standard upper ontology." In: *International Conference on Formal Ontology in Information Systems (FOIS 2001)*. Ogunquit, Maine, USA: ACM, pp. 2–9.

Novischi, Adrian and Dan Moldovan (2006). "Question answering with lexical chains propagating verb arguments." In: *21st International Conference on Computational Linguistics and the 44th Annual Meeting of the ACL (COLING/ACL 2006)*. Sydney, Australia: Association for Computational Linguistics, pp. 897–904.

Ofoghi, Bahadorreza, John Yearwood, and Ranadhir Ghosh (2006*a*). "A hybrid question answering schema using encapsulated semantics in lexical resources." In: *19th Australian Joint Conference on Artificial Intelligence (AI 2006)*. Vol. 4304/2006. Hobart, Tasmania, Australia, pp. 1276–1280.

— (2006*b*). "A semantic approach to boost passage retrieval effectiveness for question answering." In: *29th Australian Computer Science Conference*. Vol. 48. Hobart, Tasmania, Australia, pp. 95–101.

— (2006*c*). "A semantic method to information extraction for decision support systems." In: *12th Americas Conference on Information Systems (AMCIS 2006)*. Acapulco, Mexico, pp. 1475–1481.

— (2007). "A within-frame ontological extension on FrameNet: Application in predicate chain analysis and question answering." In: *20th Australian Joint Conference on Artificial Intelligence (AI 2007)*. Griffith University, QLD, Australia, pp. 404–414.

Ofoghi, Bahadorreza, John Yearwood, and Liping Ma (2007). "Two-step comprehensive open domain text annotation with frame semantics." In: *Australasian Language Technology Workshop 2007*. Melbourne, Australia, pp. 83–91.

— (2008*a*). "FrameNet-based fact-seeking answer processing: A study of semantic alignment techniques and lexical coverage." In: *21st Australasian Joint Conference on Artificial Intelligence (AI 2008)*. Auckland, New Zealand, pp. 192–201.

— (2008*b*). "The impact of semantic class identification and semantic role labeling on natural language answer extraction." In: *30th European Conference on Information Retrieval (ECIR 2008)*. Glasgow, Scotland, pp. 430–437.

— (2009). "The impact of frame semantic annotation, frame alignment techniques, and fusion methods on factoid answer processing." In: *Journal of the American Society for Information Science and Technology (JASIST)* 60.2, pp. 247–263.

Oh, Hyo-Jung, Sung Hyon Myaeng, and Myung-Gil Jang (2007). "Semantic passage segmentation based on sentence topics for question answering." In: *Information Sciences* 177.18, pp. 3696–3717.

Paşca, Marius and Sanda Harabagiu (2001). "The informative role of WordNet in open-domain question answering." In: *NAACL 2001 Workshop on WordNet and Other Lexical Resources: Applications, Extensions, and Customizations*. Carnegie Mellon University, Pittsburgh PA, pp. 138–143.

Paek, Hyung, Yacov Kogan, Prem Thomas, Seymour Codish, and Michael Krauthammer (2006). "Shallow semantic parsing of randomized controlled trial reports." In: *AMIA 2006 Annual Symposium*. Hilton Washington and Towers, Washington, DC.

Palmer, Martha, Daniel Gildea, and Paul Kingsbury (2005). "The Proposition Bank: An annotated corpus of semantic roles." In: *Computational Linguistics* 31.1, pp. 71–106.

Pazienza, Maria Teresa, Marco Pennacchiotti, and Fabio Massimo Zanzotto (2006). "Mixing WordNet, Verbnet and PropBank for studying verb relations." In: *Fifth International Conference on Language Resources and Evaluation (LREC 2006)*. Genoa, Italy.

Petruck, Miriam R.L. (1996). *Frame semantics*. Handbook of Pragmatics 1996. Philadelphia: John Benjamins.

Pradhan, Sameer, Valerie Krugler, Wayne Ward, Daniel Jurafsky, and Jim Martin (2002). "Using semantic representations in question answering." In: *International Conference on Natural Language Processing (ICON 2002)*. Bombay, India, pp. 195–203.

Pradhan, Sameer, K. Hacioglu, W. Ward, J. Martin, and D. Jurafsky (2003). "Semantic role parsing: Adding semantic structure to unstructured text." In: *Third IEEE International Conference on Data Mining (ICDM 2003)*. Melbourne, Florida, US.

Pradhan, Sameer, W. Ward, K. Hacioglu, J. Martin, and D. Jurafsky (2004). "Shallow semantic parsing using support vector machines." In: *Human Language Technology Conference/North American chapter of the Association for Computational Linguistics annual meeting (HLT/NAACL 2004)*. Boston, US.

Prager, John M., Jennifer Chu-Carroll, and Krzysztof Czuba (2001). "Use of WordNet hypernyms for answering what-is questions." In: *Tenth Text Retrieval Conference (TREC 2001)*, pp. 250–258.

Rinaldi, Fabio, James Dowdall, Kaarel Kaljurand, Michael Hess, and Diego Molla (2003). "Exploiting paraphrases in a question answering system." In: *Second International Workshop on Paraphrasing: Paraphrase Acquisition and Applications (IWP 2003)*. Sapporo, Japan: Association for Computational Linguistics, pp. 25–32.

Roberts, Ian and Robert Gaizauskas (2004). "Evaluating passage retrieval approaches for question answering." In: *Advances in Information Retrieval* 2997, pp. 72–84.

Robertson, Stephen E., S. Walker, and M. Beaulieu (1998). "Okapi at TREC-7: Automatic ad hoc, filtering, VLC and interactive." In: *Seventh Text Retrieval Conference (TREC-7)*, pp. 253–265.

Robertson, Stephen E., S. Walker, M. Hancock-Beaulieu, M. Gatford, and A. Payne. (1995). "Okapi at TREC-4." In: *Fourth Text Retrieval Conference (TREC-4)*.

Ruppenhofer, Josef, Michael Ellsworth, Miriam R.L. Petruck, and Christopher R. Johnson (2005). *FrameNet: Theory and practice*.

Rus, Vasile (2002). "Logic forms for Wordnet glosses." PhD thesis. Southern Methodist University.

Sachs, Matthew (2004). *Enhancing machine translation via frame-semantic data*. Tech. rep.

Scheffczyk, Jan, C.F. Baker, and Srini Narayanan (2006). "Ontology-based reasoning about lexical resources." In: *ONTOLEX 2006*. Ed. by A. Oltramari. Genoa, Italy, pp. 1–8.

Scheffczyk, Jan, Adam Pease, and Michael Ellsworth (2006). "Linking FrameNet to the suggested upper merged ontology." In: *2006 International Conference on Formal Ontology in Information Systems (FOIS 2006)*. Baltimore, Maryland (USA).

Schiffman, Barry, Kathleen R. McKeown, Ralph Grishman, and James Allan (2007). "Question answering using integrated information retrieval and information extraction." In: *Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT 2007)*. Rochester, New York: Association for Computational Linguistics, pp. 532–539.

Schlaefer, Nico, Jeongwoo Ko, Justin Betteridge, Manas Pathak, Eric Nyberg, and Guido Sautter (2007). "Semantic extensions of the Ephyra QA system for TREC 2007." In: *Sixteenth Text Retrieval Conference (TREC 2007)*.

Schuler, Karin Kipper (2005). "VerbNet: A broad-coverage, comprehensive verb lexicon." PhD thesis. University of Pennsylvania.

Shen, Dan and Mirella Lapata (2007). "Using semantic roles to improve question answering." In: *2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*. Prague, Czech Republic, pp. 12–21.

Shi, Lei and Rada Mihalcea (2004). "Open text semantic parsing using FrameNet and WordNet." In: *Human Language Technology Conference/North American Chapter of the Association for Computational Linguistics (HLT/NAACL 2004)*. Boston, US.

Singhal, Amit, Chris Buckley, and Ar Mitra (1996). "Pivoted document length normalization." In: *19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 1996)*. Zurich, Switzerland: ACM, pp. 21–29.

Small, Sharon, Tomek Strzalkowski, Ting Liu, Sean Ryan, Robert Salkin, Nobuyuki Shimizu, Paul Kantor, Diane Kelly, Robert Rittman, and Nina Wacholder (2004). "HITIQA: Towards analytical question answering." In: *20th International Conference on Computational Linguistics (COLING 2004)*. Geneva, Switzerland: Association for Computational Linguistics.

Stokes, Nicola, Yi Li, Lawrence Cavedon, and Justin Zobel (2007). "Exploring abbreviation expansion for genomic information retrieval." In: *Australasian Language Technology Workshop 2007*. Melbourne, Australia, pp. 100–108.

Sun, Renxu, Chai-Huat Ong, and Tat-Seng Chua (2006). "Mining dependency relations for query expansion in passage retrieval." In: *29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2006)*. Seattle, Washington, USA: ACM Press, pp. 382–389.

Sun, Renxu, J. Jiang, Y.F. Tan, H. Cui, Tat-Seng Chua, and M.-Y. Kan (2005). "Using syntactic and semantic relation analysis in question answering." In: *Fourteenth Text Retrieval Conference (TREC 2005)*.

Surdeanu, Mihai, Sanda Harabagiu, John Williams, and Paul Aarseth (2003). "Using predicate-argument structures for information extraction." In: *41st Annual Meeting on Association for Computational Linguistics (ACL 2003)*. Sapporo, Japan: Association for Computational Linguistics, pp. 8–15.

Tellex, Stefanie, Boris Katz, Jimmy Lin, Aaron Fernandes, and Gregory Marton (2003). "Quantitative evaluation of passage retrieval algorithms for question answering." In: *26th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*. Toronto, Canada: ACM Press, pp. 41–47.

Thompson, Cynthia A., Roger Levy, and Christopher D. Manning (2003). "A generative model for semantic role labeling." In: *Fourteenth European Conference on Machine Learning (ECML 2003)*. Cavtat, Croatia, pp. 397–408.

Tiedemann, Jorg (2005). "Integrating linguistic knowledge in passage retrieval for question answering." In: *Conference on Human Language Technology and Empirical Methods in Natural Language Processing*. Vancouver, British Columbia, Canada, pp. 939–946.

Vicedo, Jose (2001). "Using semantics for paragraph selection in question answering systems." In: *Eighth Symposium on String Processing and Information Retrieval (SPIRE'01)*. Chile, pp. 220–227.

Vicedo, Jose and A. Ferrandez (2001). "University of Alicante at TREC-10." In: *Tenth Text Retrieval Conference (TREC 2001)*.

Voorhees, Ellen (1994). "Query expansion using lexical-semantic relations." In: *17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 1994)*. Dublin, Ireland: Springer-Verlag New York, pp. 61–69.

— (1999). "The TREC-8 question answering track report." In: *Eighth Text Retrieval Conference (TREC-8)*, pp. 77–83.

— (2000). "Overview of the TREC-9 question answering track." In: *Ninth Text Retrieval Conference (TREC-9)*, pp. 71–80.

— (2001). "Overview of the TREC 2001 question answering track." In: *Tenth Text Retrieval Conference (TREC 2001)*, pp. 42–52.

— (2002). "Overview of the TREC 2002 question answering track." In: *Eleventh Text Retrieval Conference (TREC 2002)*.

— (2003). "Overview of the TREC 2003 question answering track." In: *Twelfth Text Retrieval Conference (TREC 2003)*, pp. 54–69.

— (2004). "Overview of the TREC 2004 question answering track." In: *Thirteenth Text Retrieval Conference (TREC 2004)*.

Voorhees, Ellen and Hoa Trang Dang (2005). "Overview of the TREC 2005 question answering track." In: *Fourteenth Text Retrieval Conference (TREC 2005)*.

Weizenbaum, Joseph (1966). "ELIZA - A computer program for the study of natural language communication between man and machine." In: *Communications of the ACM* 9.1, pp. 36–45.

Wilensky, Robert (1982). *Talking to UNIX in English: An overview of an on-line UNIX consultant*. Tech. rep.

Winograd, Terry (1972). "Procedures as a representation for data in a computer program for understanding natural language." In: *Cognitive Psychology* 3.1.

Woods, William A., R.M. Kaplan, and B.N. Webber (1972). *The lunar sciences natural language information system: Final report*. Tech. rep.

Woods, William A., S. Green, P. Martin, and A. Houston (2000*a*). "Halfway to question answering." In: *Ninth Text Retrieval Conference (TREC-9)*, pp. 489–501.

Woods, William A., Lawrence A. Bookman, Ann Houston, Robert J. Kuhns, Paul Martin, and Stephen Green (2000*b*). "Linguistic knowledge can improve information retrieval." In: *Sixth Conference on Applied Natural Language Processing*. Seattle, Washington: Morgan Kaufmann Publishers Inc., pp. 262–267.

Xu, Jinxi and W. Bruce Croft (1996). "Query expansion using local and global document analysis." In: *19th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*. Zurich, Switzerland: ACM Press, pp. 4–11.

Xue, Nianwen and Martha Palmer (2004). "Calibrating features for semantic role labeling." In: *Conference on Empirical Methods in Natural Language Processing (EMNLP 2004)*. Barcelona, Spain.

Yang, Hui and Tat-Seng Chua (2002). "The integration of lexical knowledge and external resources for question answering." In: *Eleventh Text Retrieval Conference (TREC 2002)*.

Zaanen, Menno van and Diego Molla (2007). "AnswerFinder at QA@CLEF 2007." In: *CLEF 2007 Workshop*. Ed. by Alessandro Nardi and Carol Peters. Budapest, Hungary.

Zhou, Wei, Clement Yu, Vetle I. Torvik, and Neil R. Smalheiser (2006). "A concept-based framework for passage retrieval in Genomics." In: *Fifteenth Text Retrieval Conference (TREC 2006)*.

About the Author

Bahadorreza Ofoghi
Graduate School of Information Technology and Mathematical Sciences
University of Ballarat, AUSTRALIA

Bahadorreza Ofoghi received his B.Sc. in Computer Science (Software Engineering) from Azad University, Tehran North Branch in 1999. He finished his M.Sc. degree in Computer Science (Artificial Intelligence and Robotics) at Azad University, Tehran Science & Research Branch in 2002. He worked as a computer programmer, researcher, project manager, and project supervisor at Iran Telecommunication Research Centre and other companies in Tehran during and after his studies. He has been studying his PhD at University of Ballarat, Australia since March 2005.