Interaction-based Information Retrieval in Multimodal, Online, Artefact-Focused Meeting Recordings

Matt-Mouley Bouamrane

A thesis submitted to the University of Dublin, Trinity College

in fulfillment of the requirements for the degree of

Doctor of Philosophy

April 2007

Interaction-based Information Retrieval in Multimodal, Online, Artefact-Focused Meeting Recordings

Approved by
Dissertation Committee:

Professor Alan F. Smeaton,
Dublin City University

Professor Khurshid Ahmad,
Trinity College Dublin

_____

_____

_____

_____

_____

Declaration

I, the undersigned, declare that this work has not previously been submitted to this or any other University, and that unless otherwise stated, it is entirely my own work. This thesis may be borrowed or copied upon request with the permission of the Librarian, University of Dublin, Trinity College.

Matt-Mouley Bouamrane

Dated: November 9, 2007

The Archer Metaphor

The archer metaphor: semantics of actions in remote meetings...

Figure 1: Chinese Archer, (1860s)

Photograph by John Thompson. Photo reproduced with the kind permission of Stephen Selby, of Asian Traditional Archery Research Network (ATARN), URL: http://www.atarn.org/

Acknowledgements

I would like to thank first and foremost my supervisor Dr. Saturnino Luz for all his support over the three years of this research project. He has always been generous both in time and advice and his precious opinion has spared us being side-tracked on many occasions. I would like to wish the best of luck to all the members of the Computational Linguistic Group and Machine Learning Group of the University of Dublin, Trinity College. I would particularily like to thank Proffessor Padraig Cunningham for first bringing this research position to our attention, Kenneth Brian for initiating us to his unmatched mastery of LaTeX tables and finally, Michael Carney: thanks to whose research I can safely make extravagant financial decisions while taking minimum risks...

I would like to thank all the people who participated somehow in the research project and without whom none of this would had been possible. The list is long: Dr Masood Masoodian of the University of Waikato, New-Zealand, whith whom Computer Supported Collaborative Work stopped being a wishful concept to become an Art-in-Practice. I also wish to thank David Milne, for his work on the RECOLED shared-text editor and James Glennon, Cillian Kelly and Gerrard Lynch, as their final year projects involved spending considerable hours recording and analysing meetings. Finally, thanks to all the people who somehow participated in the evaluation of the RECOLED and the Meeting Miner systems.

I would like to thank the Meeting Browser, ScanMail, Video Manga, Portable Meeting Recorder-Muvie, Teamspace-MeetingViewer, WorkspaceNavigator and Ferret development teams for kindly giving us the permission to reproduce a screenshot of these systems in the review chapter of this thesis.

Finally, I would like to thanks my familly and my friends for all their support over the years. Last by not least, I dedicate this thesis to my partner Melba for all her support and to our baby daughter Aïsha-Jane.

Matt-Mouley Bouamrane

University of Dublin, Trinity College

April 2007

Abstract

Traditional search operations, topic detection or summarisation of meeting recordings are generally performed using segmentation and indexing techniques originally developed in the field of Multimedia Information Retrieval. However, many assumptions possible in a media production environment are often inadequate when applied to spontaneous meeting recordings, as features such as sudden changes in sound energy levels or high motion can be sparse or even inexistent in typical meetings. As a result, the dominant paradigm used for meeting browsing currently consists in performing text-based information retrieval operations on automatic speech recognition (ASR) transcripts. Meetings produce however another type of information not readily available in other multimedia recordings: interactions between participants. Although there is a growing interest in using this rich source of information, it remains difficult to harness due to the current limitations of (speech, gesture or higher-level action) recognition technologies.

In computer-mediated online meetings, in which a space-based artefact (shared text or graphical document) acts as the focal point of the meeting, it is possible to generate metadata describing low-level actions of participants *during* the meeting. The semantics of these actions is many-fold: it is defined by the *person* who performed the actions, the *nature*, the *content*, the *timing* of these actions and finally the *context* (or target) of these actions. We explore a number of segmentation, indexing and search techniques specifically based on information collected about participants' actions. We developed a temporal model in which navigation of meeting recordings is performed according to actions'*content* or actions'*context*. We investigate the relationships between the *timing and content* of actions and concurrent speech communications and if the temporal distance between the content of certain actions can be used as a reliable indication of *semantic relatedness* (topic) between these neighbouring actions. We explore visualisation of meeting information centred on the concept of *data objects with persisting histories*, rather than the more traditional Multimedia concept of media streams. A meeting browsing tool called the *"Meeting Miner"* was implemented. Evaluation of the Meeting Miner was performed through an analytic evaluation, a usability study, and a task-oriented information retrieval experiment. We complement the emerging Browser Evaluation Test (BET) framework with additional performance metrics. Results of our evaluation showed that interaction-based techniques incorporated into meeting browsing systems can indeed be used efficiently for navigating multimodal meeting recordings.

Contents

The Archer Metaphor **v**

Acknowledgements **vi**

Abstract **vii**

Contents **xiii**

List of Figures **xiv**

List of Tables **xviii**

Related Publications **xx**

Glossary **xxii**

List of Symbols **xxiii**

Chapter 1 Introduction **1**

1.1 Research Motivation 1

1.1.1 Why Meetings? 1

1.1.2 Why "Artefact"-Focused? 1

1.1.3 Why Online? 2

1.1.4 Why Record Meetings? 3

1.1.5 Issues in Finding Information in Meetings 3

1.1.6 Research Motivation 3

1.1.7 Originality of this Research 4

1.2 Structure of the thesis 5

1.3 Contributions 6

Chapter 2 Meeting Browsing **9**

2.1 Speech Browsing 9

2.1.1 Speaker Segmentation . . . 10
2.1.2 Speech Skimming . . . 11
2.1.3 Automatic Speech Recognition . . . 12
2.1.4 Word Spotting . . . 13
2.1.5 Topic Segmentation . . . 14
2.1.6 Spoken Language Summarisation . . . 16
2.2 Video Browsing . . . 17
2.2.1 Visual Indexing . . . 18
2.2.2 Video Summarisation and Skimming . . . 19
2.3 Artefact Browsing . . . 20
2.3.1 Meeting Minutes Systems . . . 20
2.3.2 Implications for Browsing . . . 21
2.4 Meeting Browsers . . . 22
2.4.1 The Meeting Browser . . . 22
2.4.2 The SCAN system . . . 23
2.4.3 Video Manga . . . 24
2.4.4 The Portable Meeting Recorder — MuVie . . . 25
2.4.5 The MeetingViewer . . . 26
2.4.6 COMAP and HANMER . . . 27
2.4.7 The Ferret Media Browser . . . 28
2.4.8 WorkspaceNavigator . . . 28
2.5 Meeting Browser Evaluation . . . 30
2.6 Summary . . . 31

Chapter 3 Temporal, Specification and Content Models in Multimedia 32

3.1 Overview . . . 33
3.2 Multimedia Temporal Models . . . 33
3.2.1 Multimedia Synchronisation Issues . . . 33
3.2.2 Temporal Intervals-based Models . . . 34
3.2.3 Event-based Models . . . 40
3.3 Multimedia Specification Models . . . 41
3.3.1 HyTime . . . 41
3.3.2 MHEG . . . 41
3.3.3 SMIL . . . 42
3.4 Multimedia Content Models . . . 42
3.4.1 MPEG . . . 42
3.5 Concluding Remark . . . 44

Chapter 4 The Meeting Artefact History Model **45**
4.1 Artefact-Focused Meetings . . . 46
4.2 Generic Low-Level Interaction Metadata in Artefact-Focused Remote Meetings . . 47
4.3 Artefact History Model . . . 48
4.3.1 Key Concepts . . . 49
4.3.2 Object-based Retrieval . . . 50
4.3.3 Object Temporal Associations . . . 52
4.3.4 Action-based Browsing . . . 53
4.3.5 Advanced Query Model . . . 54
4.4 Summary . . . 55

Chapter 5 Implementation: The RECOLED Online Meeting Environment **56**
5.1 Design of a Remote Meeting Environment . . . 56
5.1.1 RECOLED: the Collaborative Development of a Collaborative Software . . 56
5.1.2 Choice of Communication Modalities . . . 57
5.1.3 Choice of Network Architecture . . . 57
5.2 Implementation of RECOLED Meeting Environment . . . 58
5.2.1 System Architecture . . . 58
5.2.2 Data capture . . . 59
5.2.3 Document consistency and Concurrency control . . . 59
5.3 RECOLED User Interface & Awareness Mechanisms . . . 61
5.3.1 Group Awareness mechanisms . . . 61
5.4 Interaction Metadata Generation . . . 64
5.4.1 Editing Primitives . . . 65
5.4.2 Timestamping . . . 65
5.4.3 Timestamping Manipulation Rules . . . 66
5.4.4 Examples of timestamps manipulation . . . 69
5.4.5 Paragraph naming scheme . . . 69
5.4.6 Paragraph Split . . . 70
5.4.7 Taxonomy of Cut operations . . . 72
5.5 Summary . . . 72

Chapter 6 RECOLED Usability Study and Meeting Corpus Collection **73**
6.1 Introduction . . . 73
6.2 Usability Study of RECOLED . . . 73
6.2.1 Methodology . . . 73
6.2.2 Usability Results and Implications . . . 75
6.2.3 Participants' feedback . . . 77

6.3 Corpus Collection . . . 77
6.3.1 Meeting Scenarios . . . 78
6.3.2 Meeting Data . . . 78
6.3.3 Corpus Description . . . 79
6.4 Summary . . . 81

Chapter 7 An Interaction-based meeting browsing tool: the Meeting Miner 83
7.1 Introduction . . . 83
7.2 Meeting Modelling . . . 84
7.2.1 Meeting Representation . . . 84
7.2.2 Semantic Persistence & Semantic Overlap . . . 86
7.3 Content-based ("by proxy") Browsing Paradigms . . . 89
7.3.1 Limitations and Strength of Interaction-based Information Retrieval . . . 89
7.3.2 Controlled Vocabulary . . . 89
7.3.3 Keywords Search . . . 90
7.3.4 Complex Query - Topic Search . . . 92
7.3.5 Action-based browsing . . . 94
7.3.6 Limitations of Content-Based Search Functionalities . . . 95
7.4 Context based Browsing . . . 96
7.4.1 Simple Paragraph-based Retrieval . . . 96
7.4.2 Extended Paragraph-based Retrieval: Temporal Neighbourhood . . . 98
7.4.3 Noise as Limitations of Time-based Contextual Information Representation 99
7.5 Meeting Miner implementation: the User Interface . . . 104
7.6 Adding Speech Recognition . . . 107
7.6.1 Speech Recognition Component . . . 107
7.6.2 Integrating ASR transcripts in the Meeting Miner . . . 109
7.7 Browsing a Meeting with the Meeting Miner . . . 111
7.7.1 Browsing with a Paragraph Temporal Neighbourhood . . . 111
7.7.2 Action-based Navigation . . . 112
7.7.3 Keyword and Topic Searches . . . 112
7.8 Conclusion . . . 112

Chapter 8 Meeting Miner Analytic Evaluation 114
8.1 Introduction . . . 114
8.2 Methodology . . . 115
8.2.1 Definition of Evaluation Metrics: Precision and Recall . . . 115
8.2.2 Meeting Evaluation Subset and the Random Samples' Approach . . . 116
8.3 Precision & Recall of Time-based Keyword Search . . . 117

8.3.1 Highest Written Frequency Keywords . . . 117
8.3.2 Least Written Frequency Keywords . . . 119
8.3.3 Random Written Frequency Keywords . . . 122
8.3.4 Results: All Keywords . . . 124
8.4 Precision & Recall of Time-based and ASR combined Keyword Search . . . 125
8.5 Topic Search Evaluation . . . 130
8.6 Concluding Remarks on Analytic Evaluation . . . 131

Chapter 9 Meeting Miner Heuristic, Usability & Task-Oriented Evaluation 133
9.1 Introduction . . . 133
9.1.1 Overview of Interactive Systems Evaluation Methods . . . 134
9.1.2 Goals of the evaluation . . . 134
9.2 Heuristic Evaluation . . . 135
9.2.1 Methodology . . . 136
9.2.2 Results of the Heuristic Evaluation . . . 136
9.2.3 Meeting Miner functionalities rated satisfactorily . . . 138
9.2.4 Recommendations which were implemented (or not) in the Meeting Miner . . . 139
9.2.5 Concluding Remarks on Heuristic Evaluation . . . 142
9.3 Pilot Usability Study . . . 143
9.3.1 Methodology . . . 143
9.3.2 Results of Usability Evaluation . . . 143
9.3.3 Concluding Remarks on Usability Evaluation . . . 147
9.4 Browser Evaluation Test - Information Retrieval Task . . . 147
9.4.1 Goals of the Information Retrieval Task . . . 147
9.4.2 Methodology . . . 148
9.4.3 Evaluation Metrics . . . 152
9.4.4 Proposed Interpretation of Novel Metrics . . . 157
9.4.5 Results of the IR Task . . . 159
9.4.6 Concluding Remarks on Information Retrieval Task . . . 164
9.5 Final Usability Study . . . 164
9.5.1 Interactions with The Meeting Miner . . . 165
9.5.2 Participants' Comments . . . 166
9.5.3 Strength & Limitations of Current System . . . 169
9.6 Concluding Remarks on the Meeting Miner Evaluation . . . 171

Chapter 10 Conclusion **173**

10.1 Validation of Hypothesis 173

10.2 Lessons Learned 174

10.3 Future Research 175

Bibliography **177**

Appendix A RECOLED Usability Questionnaire **190**

Appendix B ASR Speech Transcripts of a Meeting **192**

Appendix C Results of Analytic Evaluation of Topic Search **195**

Appendix D Heuristic Evaluation Report **197**

Appendix E MeetingMiner Usability Questionnaire **199**

Appendix F Questions of the Information Retrieval Task **201**

List of Figures

1 Chinese Archer, (1860s) v

1.1 Meeting, an ubiquitous and pervasive human activity...
(Picture reproduced with permission of logo owner: Blaggard's Pub, New York) . 2
1.2 A Multidisciplinary Research where many disciplines of Computer Science converge 7

2.1 The Meeting Browser user interface 22
2.2 The SCAN user interface 23
2.3 A meeting summary produced by Video Manga 25
2.4 The Muvie client user interface 26
2.5 The MeetingViewer User Interface 27
2.6 Hanmer User Interface 28
2.7 The Ferret Media Browser 29
2.8 WorkspaceNavigator User Interface 29

3.1 Temporal Relations between two media intervals (according to Allen (1983)) . . . 35
3.2 Enhanced Intervals Relations between two media items (adapted from Wahl and Rothermel (1994)) 36
3.3 Simple Marked Petri Net 37
3.4 Timed Petri Nets and corresponding Allen Interval representations (according to Little and Ghafoor (1990)) 38
3.5 Standard (a.) v.s Fuzzy (b.) Temporal Interval Representations 39
3.6 Event based temporal view of multimedia document (adapted from Buchanan and Zellweger (1992)) 40

4.1 Two examples of artefacts produced as outcomes of collaborative activities 46
4.2 The Archer & Target metaphor for interactions in remote meetings 48
4.3 The data stream or timeline perspective 50
4.4 Data Object perspective 51
4.5 Data Objects Temporal Associations 52

4.6 Sequence of Actions . . . 53
4.7 Using non-verbal actions as a navigation tool . . . 54

5.1 RECOLED's Client-Server system architecture . . . 58
5.2 RECOLED's audio and editing history recording architecture . . . 59
5.3 The concurrency control mechanism of RECOLED . . . 61
5.4 RECOLED client application interface . . . 62
5.5 RECOLED Client avatars and their actions . . . 62
5.6 Telepointing on a paragraph of the Meeting Document . . . 63
5.7 Free hand drawing . . . 63
5.8 Gesture indicators . . . 64
5.9 RECOLED paragraph timestamps . . . 66
5.10 RECOLED action timestamps . . . 67
5.11 Moving the textual content *"text1\n"* from first to third paragraph position in the document structure . . . 68
5.12 Effect of paragraph content manipulation on paragraph timestamps . . . 68
5.13 Effect of merging several paragraphs on paragraph timestamps . . . 69
5.14 RECOLED paragraph naming scheme . . . 70
5.15 Two possible timestamping manipulation rules in the "Paragraph Split" case . . . 70

6.1 Meeting Recording Architecture . . . 78
6.2 Document type definition for collaborative meeting document . . . 80

7.1 Three possible Meeting Representations . . . 84
7.2 Representation of Semantic Anticipation, Persistence & Overlap using fuzzy temporal intervals . . . 87
7.3 Strict v.s. Extended Temporal Interval Overlap Relationships . . . 88
7.4 The concurrency of typed and spoken keywords assumption: the assumption (represented by arrow) is only made from Text-to-Speech . . . 91
7.5 The Topic Assumption: time distance between occurences of words in a Topic Search $\mathcal{TS}_{and}$ $(\mathcal{K}w_a, \mathcal{K}w_b)$ is used as a measure of semantic likelihood between these keywords. 94
7.6 Using the content of Edits as an audio navigation tool . . . 94
7.7 A numbering change in the course of the meeting can result in potentially misleading Keyword searches . . . 95
7.8 Using contextual information for accessing a meeting recording . . . 96
7.9 Paragraph history permits to track down contextual modifications such as labelling changes . . . 97
7.10 Paragraph Temporal Neighbourhood Retrieval . . . 98

7.11 A paragraph Temporal Neighbourhood 99
7.12 Paragraph association highlighted in the meeting outcome through a paragraph-based temporal neighbourhood retrieval 100
7.13 Contextual Information Overload 100
7.14 The high level of verbal acknowledgements in remote meetings 101
7.15 Speech intervals amalgamation for compact graphical representation of contextual information 102
7.16 The icons used for items and operations descriptions in the user interface, (Open Source: http://wm-icons.sourceforge.net/) 102
7.17 Paragraph Temporal Neighbourhood with information redundancy removed 103
7.18 Raw content of edits and extracted action keywords 103
7.19 Paragraph Temporal Neighbourhood as speech nodes with associated contextual information 104
7.20 Earlier version of Meeting Miner User Interface 105
7.21 Meeting Miner User after modifications performed as the results of the usability evaluation 106
7.22 Earlier version of Meeting Miner including Speech Recognition transcripts 109
7.23 Excerpt of participants ASR transcripts (in color) and the corresponding true speech (in grey) 110

8.1 Word Spotting Sets, the figures represent a percentage of the combined number of words correctly spotted by the ASR and Time-based techniques 126
8.2 Precision results for combinations of Time-based and ASR-based word spotting techniques 129
8.3 Estimation of Precision and Recall for Time-based only and combined ASR and Time-based word spotting (WER=39.4%) 129

9.1 Three modes of displaying index terms on the audio view: (i) Term Frequency, (ii) Inverse Term Frequency or (iii) selecting a specific term 140
9.2 Greater audio navigation options introduced as a result of the heuristic evaluation 141
9.3 Flexibility of Topic Search: AND & OR mode, which are explicitly displayed in the user interface 141
9.4 Participants' ranking (according to usefulness) of components and functionalities of the Meeting Miner 144
9.5 Participants' first preference choice in the Meeting Miner functionalities 144
9.6 The Random Access Interface 150
9.7 The Meeting Questionnaire User Interface 151
9.8 The IR task experiment: Meeting Miner and Questionnaire User Interface 152

9.9 Information Accumulation and First Correct Answer-Time (FCA-t) 154
9.10 Critical Information Distribution and Information Span of evaluation meeting A . 154
9.11 A meeting Information Span depends on the focus of the information retrieval task 155
9.12 A User's Information Accessed Ratio of evaluation meeting A 155
9.13 Sequential listening may result in relatively good score at the task but with low Information Accessed Ratio 156
9.14 Spanned Recall for 2 types of browsing behaviour 157
9.15 Recall according to Participants & interfaces' conditions, with Meeting A on the left and Meeting B on the Right 161
9.16 Spanned Recall according to participants & interfaces' conditions, with Meeting A on the left and Meeting B on the Right 164
9.17 Participants' interactions with the various components of the Meeting Miner during the IR task 165
9.18 Query Reformulation Mechanism using a Thesaurus, Controlled Vocabulary & User feed-back 172

A.1 RECOLED Usability Questionnaire: page 1 of 2 190
A.2 RECOLED Usability Questionnaire: page 2 of 2 191

B.1 ASR Speech transcripts of 1 hour long meeting with 2 participants: first 2 pages . 192
B.2 ASR Speech transcripts of 1 hour long meeting with 2 participants: page 1 to 12 . 193
B.3 ASR Speech transcripts of 1 hour long meeting with 2 participants: page 13 of 23 . 194

D.1 Form used for Heuristic Evaluation: page 1 of 2 197
D.2 Form used for Heuristic Evaluation: page 2 of 2 198

E.1 Usability Questionnaire: page 1 of 2 199
E.2 Usability Questionnaire: page 2 of 2 200

List of Tables

5.1 Taxonomy of Cut operations and their effect on paragraph timestamps 71
5.2 Break down of a single cut operation into the types defined in Table 5.1 72

6.1 Corpus composition according to task 80

7.1 Word Error Rate obtained for the transcripts of meeting evaluation subset 109

8.1 Description of meeting evaluation subset 117
8.2 Precision, Recall and average listening time of keyword search for Highest Written Frequency Keywords 118
8.3 Precision, Recall and average listening time of keyword search for Least Written Frequency Keywords 119
8.4 Precision, Recall and average listening time for 50 random keywords over five samples of the meetings corpus 123
8.5 Precision, Recall and average listening time for 100 random keywords of random frequency over five samples of the meetings corpus 125
8.6 Estimation of Precision and Recall for Time-based only and combined ASR and Time-based word spotting (WER=39.4%) 129

9.1 Evaluation methods 135
9.2 Interfaces used for IR evaluation according to each experimental condition 152
9.3 Display of results under the four experiment conditions 159
9.4 Recall And precision Results of IR task 160
9.5 First Correct Answer & Average Answering Time of IR task 162
9.6 Information Accessed Ratio & Spanned Recall 163

C.1 Topic Search results according to the Task-Oriented users' queries on meeting A . 195
C.2 Topic Search results according to the Task-Oriented users' queries on meeting B . 196
C.3 Topic Search results aummary on meetings A & B 196

F.1 Questions of the Information Retrieval Task for Meeting A, part I 201

F.2 Questions of the Information Retrieval Task for Meeting A, part II 202
F.3 Questions of the Information Retrieval Task for Meeting B, part I 203
F.4 Questions of the Information Retrieval Task for Meeting B, part II 204

Related Publications

Journal Articles

An Analytical Evaluation of Search by Content and Interaction Patterns on Multimodal Meeting Records

Matt-Mouley Bouamrane and Saturnino Luz

in Multimedia Systems Journal, Semantic Media Adaptation and Personalization Special Issue, Marios C. Angelides, Phivos Mylonas and Manolis Wallace eds.,

Volume 13, Number 2 / August, 2007, Springer Berlin / Heidelberg, p 89-102

Meeting Browsing, a State-of-the-Art Review

Matt-Mouley Bouamrane and Saturnino Luz

in Multimedia Systems Journal, User-Centered Multimedia Special Issue

Boll S. and Westermann U. eds., Volume 12, Numbers 4-5 , March, 2007, Springer Berlin / Heidelberg, p 439-457.

Conference Proceedings

In Search of a Better BET, novel metrics for a Browser Evaluation Test

Matt-Mouley Bouamrane and Saturnino Luz

in Proceedings of Ist International Conference on Theory of Information Retrieval,

ICTIR'07, Oct 2007, Alma Mater , Budapest, Hungary, p 37-50.

An Analysis of the Effectiveness of Temporal Mapping and Speech Recognition for Content-Based Multimedia Indexing

Matt-Mouley Bouamrane and Saturnino Luz

in Proceedings of Ist International Workshop on Semantic Media Adaptation and Personalization, SMAP06, Dec 2006, Athens, Greece, IEEE Computer Society Digital Library, p 1-6.

Temporal Mining of Recorded Collaborative Production of Artefacts

Matt-Mouley Bouamrane and Saturnino Luz

in Proceedings of Industrial Conference on Data Mining,

ICDM'2006, Leipzig, Germany, July 2006, p 187-201.

Navigating Multimodal Meeting Recordings with the Meeting Miner

Matt-Mouley Bouamrane and Saturnino Luz

in Proceedings of Flexible Query Answering Systems, FQAS'2006, Milan, Italy,

Lecture Notes in Artificial Intelligence, LNAI vol 4027 / June 2006,

Larsen H., Pasi G., Ortiz-Arroyo D., Andreasen T. and Christiansen H. eds.,

Springer-Verlag, p 356-367.

Gathering a Corpus of Multimodal Computer-mediated Meetings with Focus on Text and Audio Interaction

Saturnino Luz, Matt-Mouley Bouamrane and Masood Masoodian

in Proceedings of the fifth international conference on Language Resources and Evaluation,

LREC06, Genoa, Italy, May 2006, p 407-412.

Exploring the Structure of Media Stream Interactions for Multimedia Browsing

Saturnino Luz and Matt-Mouley Bouamrane

in Proceedings of Adaptive Multimedia Retrieval, AMR05, Glasgow, Scotland,

Lecture Notes in Computer Science, LNCS vol 3877 / February 2006,

Detyniecki M., Jose J., Nürnberger A. and Rijsbergen C.J.K. eds., Springer-Verlag, p 79-90.

History-based Visual Mining of Semi-structured Audio and Text

Matt-Mouley Bouamrane, Saturnino Luz and Masood Masoodian

in Proceedings of Multimedia Modelling, MMM06 Beijing, China,

IEEE Press, January 2006, p 360-363.

Supporting Remote Collaboration Through Structured Activity Logging

Matt-Mouley Bouamrane, Saturnino Luz, Masood Masoodian, and David King

in Proceedings of 4th International Conference on Grid and Cooperative Computing, GCC 2005, Beijing, China

Lecture Notes in Computer Science, LNCS vol 3795 / November 2005,

Hai Zhuge and Geoffrey C. Fox eds., Springer-Verlag, p 1096-1107.

RECOLED: A Group-Aware Collaborative Text Editor for Capturing Document History

Masood Masoodian, Saturnino Luz, Matt-Mouley Bouamrane and David King

in Proceedings of WWW/Internet 2005, Lisbon, Portugal, vol1, October 2005, p 323-330.

A Framework for Collaborative Writing with Recording and Post-Meeting Retrieval Capabilities

Matt-Mouley Bouamrane, David King, Saturnino Luz and Masood Masoodian

in Special issue on the 6th International Workshop on Collaborative Editing Systems, November 2004, IEEE Distributed Systems Online, 6 pp.

Glossary

AMI	Augmented Multiparty Interaction
ASR	Automatic Speech Recognition
BET	Browser Evaluation Test
CSCW	Computer Supported Collaborative Work
HCI (or CHI)	Human-Computer Interaction
HMM	Hidden Markov Models
IR	Information Retrieval
MMiner	Meeting Miner
MIR	Multimedia Information Retrieval
OTV	Overlap Tolerance Value
OCV	Out-of-Controlled Vocabulary term
OOV	Out-Of-Vocabulary term
RAI	Random Access Interface
RECOLED	Recording Collaborative Editor
RTP	Real-Time Protocol
STF	Spoken Term Frequency
Std. Dev.	Standard Deviation
Std. Err.	Standard Error
TF-IDF	Term Frequency - Inverse Document Frequency
TF-ITF	Term Frequency - Inverse Term Frequency
UI	User Interface
WER	Word Error Rate
WTF	Written Term Frequency
XML	Extensible Markup Language

List of Symbols

$\mid Set \mid$	Cardinal, (number of elements) in Set
Diff (str_1, str_2)	the number of differences between the strings str_1 and str_2.
$\mathcal{E}(X)$	**Expected value** (mean) of the random variable X
$\mathcal{K}S$ ( )	**Keyword Search** function
$\mathcal{K}w$	**Keyword**
$\mathcal{L}$	**List** of stop words
$\mathcal{O}$	meeting document **Outcome**
π	**Precision**
ρ	**Recall**
r	**Ratio**
σ	Standard Deviation
Σ	**Sum**
$\mathcal{S}$	The set of *all* **Speech** turns (time intervals) in a meeting
s_i	The i^{th} speech segment in $\mathcal{S}$
t	Time
$\mathcal{T}$	The set of *all* **Text** (and Gesture) events (time intervals) in a meeting
τ	Temporal threshold beyond which semantic likelihood betwen query terms is no longer inferred
T_j	The j^{th} text (or gesture) event in $\mathcal{T}$
$Trans(s_i)$	The speech recognition transcripts of speech sentence s_i
$Trans_{\mathcal{V}}(s_i)$	Speech transcripts limited to the terms contained in the controlled Vocabulary $\mathcal{V}$
$\mathcal{T}S$ ( )	**Topic Search** function
$\mathcal{V}$	controlled **Vocabulary**
$Words(s_i)$	The set of exact (spoken) words contained in the speech sentence s_i,

Chapter 1

Introduction

1.1 Research Motivation

1.1.1 Why Meetings?

The complexity of many projects performed in the workplace means that most tasks need to be carried out on a daily basis by teams involving people with various responsibilities and fields of expertise, sometimes residing in different places. Phases of individual work are punctuated by meetings of some sort to discuss progress, share ideas, take decisions, allocate tasks, etc. Meetings are thus a central aspect of human activity, pervading many aspects of our professional and social lives (Figure 1.1).

1.1.2 Why "Artefact"-Focused?

Most meetings in the workplace will involve the use of some physical documents (or perhaps their digital representations), such as slide presentations, posters, textual documents, summaries, notes or sketches on a black-board, etc. These meeting *"artefacts"* are often mentioned or presented during the course of a meeting to support the decision-making process. However, in many cases, which we will refer to as *artefact-"focused" meetings*, the main purpose of the meeting is in fact the *collaborative production* of some *meeting artefacts*, which becomes the *goal* or focal point of the meeting. Studies have shown that a majority of text documents are written involving teams of collaborators (Posner and Baecker 1992) and indeed, many meetings in real-life scenarios revolve around the production of some sort of meeting artefacts: work-plans, posters, technical drawings, architectural plans, presentations, and the endless number of projects and activities which fall under the remit of *collaborative design* (Anupam and Bajaj 1994, Bafoutsou and Mentzas 2002, van Leeuwen 2003).

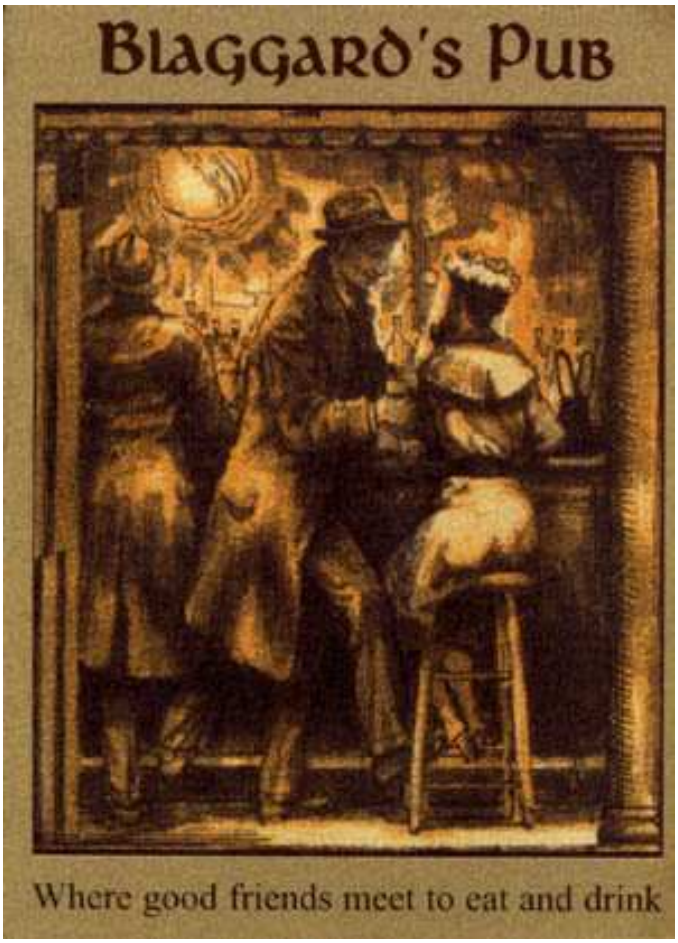


Figure 1.1: Meeting, an ubiquitous and pervasive human activity...
(Picture reproduced with permission of logo owner: Blaggard's Pub, New York)

1.1.3 Why Online?

Computers have evolved from simply being processing machines to becoming ubiquitous communications tools. The advent of broadband has considerably increased the ability to transfer extremely large volumes of digital data. Teleconferencing, once only reserved for members of the research community or corporate companies due to the need for considerable resources, is now easily accessible through low cost equipment (e.g. web cams) and widely, and in some cases freely, available software. Indeed, the recent and much reported commercial successes of providers of VoIp (Voice over IP) has revolutionised telecommunications, enabling us to communicate with people on the other side of the planet for the price of a local call. Similarly, websites like BEBO$^{\text{TM}}$, particularly popular with young users, show the increasing potential of the Internet for creating *online communities*. The combined factors of the adoption of computers as a main communication medium, coupled with lower costs and an increase in speed and bandwidth, will without doubt result in an ever increasing volume of synchronous meetings being held on the Internet in the future.

1.1.4 Why Record Meetings?

As computers become ubiquitous tools for communication, possibilities for synchronous collaboration have been greatly enhanced and the complete capture of meetings could in principle free participants from distracting and time consuming tasks such as note taking and minute production. Indeed, there are many reasons why one would want to capture and archive meetings. An internet survey carried out by Jaimes et al. (2004) and involving more than five hundred respondents sheds some light on some of the many reasons for storing and reviewing past meetings: to keep accurate records, check the veracity and consistency of statements and descriptions, revisit portions of the meeting which were misunderstood or not heard, re-examine past positions in the light of new information, obtain proofs and recall certain ideas are cited as the mains reasons.

1.1.5 Issues in Finding Information in Meetings

Recording meetings only solves part of the problem. As the number of recorded meetings grows, so does the complexity of extracting meaningful information from such recordings. Continuous media such as audio and video are difficult to access for lack of natural reference points. Navigation in these media is time consuming and can be confusing. Summarisation is a non-trivial process. A study of users browsing and searching strategies when accessing voicemail messages, sometimes of very short duration (30s), showed that people had serious problems with local navigation of messages and difficulties remembering message content (Nakatani et al. 1998). Many users performed time-consuming sequential listening of messages in order to find relevant information and often reported taking notes to remember content. In contrast, users displayed improved browsing performance, playing less audio when speech recognition transcripts were available as audio indexes in the user interface (Hirschberg et al. 1999).

1.1.6 Research Motivation

Ever increasing volumes of recorded meeting data are driving the need for the implementation of tools to efficiently access and quickly retrieve important pieces of meeting information. This is a non-trivial task for the lack of obvious structure in multimedia recordings, compounded by rich and often orthogonal communication modalities. Thus, the study of multimodal meetings currently attracts considerable interest among a wide variety of research communities (Bouamrane and Luz 2007b). In order to exploit the information contained in multiple media and provide structured access to meeting information, implementers of meeting browsing systems have typically used, adapted and combined techniques developed in the fields of digital signal processing, recognition technology, multimedia information retrieval (MIR), natural language processing (NLP), knowledge representation and human-computer interaction. However, identifying which techniques can

usefully be integrated in a meeting browser is a daunting task, especially if one wishes to take particular care in implementing a system whose whole functionality is greater than the sums of its' components. Many state-of-the art techniques successfully used in the field of multimedia information retrieval (scene detection in feature films, automatic indexing in sports or broadcast news) (Aigrain et al. 1996, Smeaton 2001, Snoek and Worring 2005) are often inappropriate when applied to spontaneous meeting recordings, which will typically lack the salient features (sudden changes in sound energy levels, high motions, scene transitions) and rigid semantic often present in a media production environment. As a result, the dominant paradigm used for meeting browsing currently consists in performing text-based information retrieval operations on Automatic Speech Recognition (ASR) transcripts (Koumpis and Renals 2005). However, summarisation and topic detection techniques successfully implemented in the field of text information retrieval can have mitigated results when applied to speech transcripts, either because of significant Word Error Rates (WER) or simply due to the fundamentally different nature of written text and spontaneous speech, as the latter will typically contain a prohibitively high number of disfluencies, repetitions and false-starts.

The limitations in deploying well tested techniques of other disciplines to the field of meeting browsing suggests the need to spend considerable efforts in identifying and understanding what is specific about meeting information, and subsequently develop appropriate models to describe this information. A key aspect in the essence of meetings which has previously been overlooked by many researchers is inherently present in meeting participants' interactions. Although there is a growing interest in using this rich source of information as a browsing modality, it remains difficult to harness because of the current limitations of (speech, gesture or higher-level action) recognition technologies. McCowan et al. (2005) are a notable exception in that they have investigated the use of low-level audio and visual features (speech activity, energy, pitch and speech rate, face and hand blob) to model meetings as a continuous sequence of high-level meeting group actions (monologue, presentation, discussion, etc) using a Hidden Markov Model based on the interactions of individual participants.

1.1.7 Originality of this Research

Our own approach to accessing information in meeting recordings strongly focuses on participants' non verbal interactions, yet it is somehow different from the previous work. While McCowan et al. essentially use recognition technology in a meeting room environment to identify high-level meeting events, our work has focused on capturing low-level non-verbal interaction in online meetings, whose focus is the collaborative production of meeting artefacts (text documents, work-plans, tables, etc.) In these scenarios, much of the information about the decision making process is contained in the continuous, audio-visual medium. An aspect of the problem that is often neglected is the relationships between timing patterns in speech (e.g. speech turns) and non-verbal actions (e.g. pointing, editing, drawing). This thesis presents an approach to meeting browsing

based on a retrieval model which builds on the structure of participants' non-verbal interactions (editing and gesturing actions) (Chapter 3, Chapter 7). A online collaborative meeting environment called RECOLED (RECording COLlaborative EDitor) was specially developed for the purpose of generating participants' actions metadata in real-time *during* the meeting (Chapter 5). RECOLED was collaboratively developed by David Milne (formerly King) and Dr. Massood Massodian of Waikato University, New Zealand, and by Dr. Saturnino Luz and the author, of the University of Dublin, Trinity College, Ireland. RECOLED (meeting environment), RECPLAJ (recording server) and REXPLORE (reference library) have been released as free software. Further information can be found in the project's web site (COWRAT 2006).

A meeting browser system called the Meeting Miner was subsequently designed by the author (Chapter 7). The Meeting Miner primarily uses action metadata, complemented by ASR transcripts, to navigate meeting recordings. This system emphasises semantic relationships between discrete interactions and the continuous (speech) medium. It also seeks to go beyond this simple "interaction-to-speech" mapping by aggregating interactions into larger semantic entities sharing the same "context" (co-occurrence or co-location within a meeting document). While the retrieval model behind the Meeting Miner builds on a number of "empirical assumptions", an analytic evaluation of the query-based search methods (Chapter 8) and an iterative heuristic, usability and task-oriented evaluations of the browser (Chapter 9) vindicated the validity of the model in practical tasks, while also exposing some its' limitations.

1.2 Structure of the thesis

This thesis is organised as follows:

Chapter 1 - Introduction.

Chapter 2 - Meeting Browsing. This chapter is a state-of-the-art review of the field of Meeting Browsing. Existing techniques for speech, video and meeting "artefacts" browsing are covered in detail. Existing multimodal meeting browsing tools as well as various browsers' evaluation methodologies are also introduced.

Chapter 3 - Temporal, Specification and Content Models in Multimedia. This chapter is an introduction to a number of important concepts in Multimedia modelling which are recurrent throughout this thesis, including Temporal Interval Models and other important Multimedia content description models.

Chapter 4 - The Meeting Artefact History Model. In this chapter, we proposes a novel multimedia model specifically targeted at computer-mediated meetings which result in the

collaborative production of meeting *artefacts*. This constitutes the main theoretical foundation of this thesis.

Chapter 5 - RECOLED Online Meeting Environment. This chapter presents RECOLED, an integrated architecture for capturing online collaborative meetings, supporting speech, text, and "gesturing" activities. Automatic generation of interaction metadata is described as well as manipulation rules of history logs of textual artefacts during the course of a meeting.

Chapter 6 - RECOLED Usability Study and Meeting Corpus Collection. This chapter presents a usability study of the RECOLED remote meeting environment, followed by a description of our multi-modal meetings corpus collection.

Chapter 7 - The Meeting Miner. This chapter presents how interaction information is used for providing novel navigation modalities in meeting recordings. The Meeting Miner, an interaction-based meeting browsing tool is described.

Chapter 8 - Analytic Evaluation of the Meeting Miner. This chapter describes an analytic evaluation of the search functionalities of the Meeting Miner using statistical methods and standard Information Retrieval metrics.

Chapter 9 - Heuristic, Usability & Task-Oriented Evaluation of the Meeting Miner. In this final chapter, we present an evaluation of the Meeting Miner from a user-centred perspective. The evaluation is conducted along three directions: a heuristic evaluation, two usability studies, and finally, a task-oriented evaluation task similar to the one suggested by Wellner et al. (2005) in their Browser Evaluation Test (BET). In addition, we propose several novel metrics to complement the BET.

Chapter 10 - Conclusion. A discussion on the implications of the results of this research work and directions for future work.

1.3 Contributions

Figure 1.2 illustrates how the multidisciplinary research described in this thesis is at the converging point of many disciplines of Computer Science.

Contributions to Multimedia Modelling were made in our proposal for a formal meeting artefact history model (Bouamrane and Luz (2006b), Chapter 4), in which meeting information is centred on the concept of *objects with persisting histories*, rather than the more traditional concept of media streams. Design choices behind navigation methods for meeting recordings based on interaction

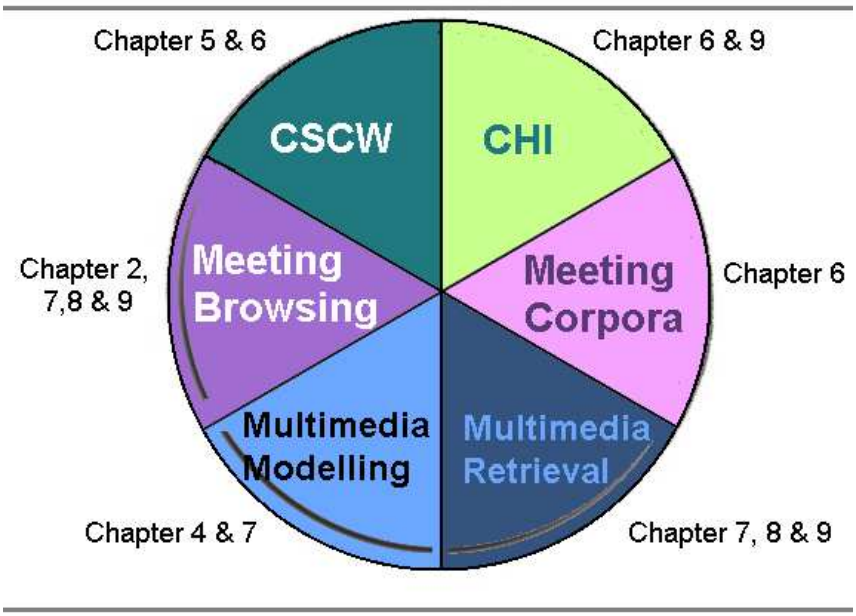


Figure 1.2: A Multidisciplinary Research where many disciplines of Computer Science converge

information is also supported by formal concepts described in Chapter 7 (Bouamrane and Luz 2006a, 2007a). These include paragraph-based temporal neighbourhoods 7.4.2 and interaction semantic persistence, which we modelled using fuzzy temporal intervals in 7.2.2.

Contributions to CSCW (Computer Supported Collaborative Work) were made during the development of the RECOLED online meeting environment. Managing interaction logs during real-time document editing proved more complex than anticipated and led us to design formal information manipulation rules (Bouamrane et al. (2005), Chapter 5), in order to maintain histories about the document creation process.

There were several contributions to the field of meeting browsing: first, in the form of providing researchers in the field with additional resources. A significant effort was made to review the state of the art (Bouamrane and Luz (2007b), Chapter 2), which we hope will become a useful resource for those with an interest in Multimedia information retrieval and meeting browsing.

In Chapter 7, we present how interaction information collected during meetings can used to develop novel navigation modalities in meeting recordings. Although, participants' activities had already been used in the past by Geyer et al. (2001) as visual indexes in meeting recording, to the best of our knowledge, this is the first time that interaction information has been used as the central information retrieval component of a meeting browsing tool.

Contributions to Multimedia Information Retrieval are to be found in the analytic evaluation of the Meeting Miner (Chapter 8). We showed that using the content of text events is indeed a very reliable way of accessing speech. We also found that interaction-based techniques exhibited a higher Recall than could be expected from a method operating on "sparse" events. We showed that this is mainly due to psycho-linguistic phaenomena, namely *linguistic convergence* and *repetitions*

in discourse (Tannen 1989). To the best of our knowledge, this is the first time that the potential implications of these phaenomena in terms of information retrieval from meeting recordings are highlighted. Finally, we have also identified limitations to term-based queries in meeting recordings (entity "re-labelling") and proposed a solution to circumvent this type of issues (Chapter 7).

Finaly, a contribution was made in the form of a comprehensive user evaluation of the Meeting Miner browsing tool (Chapter 9) which involved more than 36 distinct subjects. We describe the experience gained through developing functionalities to enhance users' browsing experience, requirements regarding information feed-back and the importance of flexibility in the browsing tool. This study of users' browsing behaviour and performance in practical tasks will provide system developers with a precious insight into usability requirements of meeting browsers. Finally, a significant element of the Meeting Miner evaluation was performed through a task-oriented evaluation. The evaluation was based on the BET (Browser Evaluation Test) framework first proposed by Wellner et al. (2005). We significantly expanded the scope of the BET evaluation framework by introducing novel performance metrics and hope that our evaluation will be useful to other researchers interested in generic evaluation methods of multimodal meeting browsers.

Chapter 2

Meeting Browsing

To be truly effective, conferencing capture systems need to offer users efficient means of navigating recordings and accessing specific information. Thus, there has been growing research interest in producing applications for navigation of multimodal meeting recordings in order to support users' information needs. Interaction modalities used in meetings and thus the nature of the recorded media will typically be dictated by the meetings' physical setup: a purpose-built meeting room (Chiu, Kapuskar, Wilcox and Reitmeier 1999, Lee et al. 2004) or Internet-based environments (Cutler et al. 2002, Erol and Li 2005)) and meeting capture capabilities.

In what follows, we review the state-of-the-art in multimodal meeting browsers. We will use the taxonomy introduced by Tucker and Whittaker (2005) where browsers are classified according to the focus of the browsing task, or primary mode used for the meeting data presentation. The three main browser categories are: *audio* browsers, *video* browsers and *artefact* browsers. The first two categories focus on communication modalities used in meetings and the contents they convey, while the third focuses on objects produced or manipulated during the meeting, such as notes, slides, drawings, plans etc. We first review the state of the art in the various existing segmentation, indexing and searching techniques for speech and video data. We then present some of the main existing multimedia and meeting browser applications, and discuss evaluation methods for such applications. An extended version of this chapter was published in (Bouamrane and Luz (2007b).

2.1 Speech Browsing

Unlike space-based media such as text and images, where one can quickly visually scan a page of text or a set of picture thumbnails to get a general impression of a document's content, audio is a medium that does not lend itself well to visualisation. Audio recordings can be of very long duration and multimedia databases may contain large numbers of such recordings. Accessing specific parts of audio documents is therefore particularly challenging because one often does not know *what* constitutes relevant information or *where* it is until one has actually heard it. Listening to entire

audio recordings is however extremely time consuming and in some cases simply not feasible. In what follows, we describe the state-of-the art in techniques for structured speech browsing. We define structured audio as the acoustic signal supplemented by an abstract representation which provides an overview of the recording, indications on the nature or importance of specific parts of the audio, and access to any location within the recording. Other comprehensive surveys of audio and speech access techniques can be found in (Foote 1999, Goldman et al. 2005, Koumpis and Renals 2005).

2.1.1 Speaker Segmentation

Visually representing audio in a meaningful manner is a particularly difficult task as there is no obvious or intuitive way of doing so. Graphically displaying an audio recording as a waveform would generally be inappropriate since, for most users, the audio signal spectrum offers no information about content. Some level of structuring can however be attained by common signal processing techniques. A frequently used strategy is to visually segment a meeting's audio track according to participants contributions over time. This technique is known as *speaker segmentation* (Hindus and Schmandt 1992, Luz and Roy 1999). When audio of various participants is recorded on a single track, speaker identification needs to be carried out prior to speaker segmentation. Speaker identification is the process of automatically distinguishing between participants' voices in order to determine when the various talkers are active (Wilcox et al. 1994). Audio browsers based on speaker segmentation will typically display a visual representation of talk spurts as a horizontal bar over a time line, identifying participants through thumbnail pictures, colours etc. Clicking on a bar will play the corresponding audio segment. A user can choose to listen to neighbouring audio segments or specific contributions.

There are a number of limitations to speaker segmentation as a browsing modality. First of all, typical meetings will contain hundreds of speech exchanges, the majority of which have very short duration. In order to visually distinguish between contributing speech sources, speaker segments will be represented on windows of short duration (e.g. a few minutes) whose time scales will be stretched in comparison to the overall meeting. Browsing through the audio file therefore implies scrolling across a large amount of these audio segment windows. This can be confusing and might make it difficult for a user to develop a clear picture of the structure of the audio recording. The other limitation of speaker segmentation lies in the fact that individual contributions may be rendered meaningless without the context (other participants' contributions) in which they were said. In accordance with the natural structure of discourse, it is reasonable to assume that audio segments in close time proximity are relevant to one another. This phenomenon can be interpreted in terms of the question-answer pair paradigm (Waibel et al. 2001) where adjacent speech exchanges are considered more informative jointly than in isolation. Therefore, segmenting conversations by topic seems a more appropriate choice for a general audio browsing task. However, speaker

segmentation can be useful when additional information is available, such as a textual description of playback points, which can precisely identify the context of specific audio contributions. Roy and Malamud (1997) have developed a system which maps text transcripts of proceedings of the United States House of Representatives to speaker transition in the audio recordings. The transcripts are manually drafted in real time during the House's sittings by a human transcriber and later edited. Participants' precise contributions regarding a particular issue can be pin-pointed by a text-to audio alignment system which provides pointers to an audio database containing hundreds of hours of recording. When selecting a portion of text from the transcripts, a user is presented with a list of audio contributors to which they can listen.

2.1.2 Speech Skimming

Various time and frequency based signal processing techniques can be applied to an acoustic signal in order to alter the play-back rate of an audio recording (Arons 1992). Playing audio at a faster rate (time compression) will thus permit a user to listen to more in less time. There is of course a limit to play-back rate increase before audio becomes unintelligible. SpeechSkimmer (Arons 1997) is a system which combines various speech processing techniques in order to navigate through audio recordings. The user can adjust the speed at which he listens to the recording: slower or faster than the normal rate. Playing the audio at an increased speed is achieved through time compression techniques involving sampling of the original signal, while the entire audio content itself is preserved. Skimming, on the other hand, involves playing only selected sections of the recording. Selection is based on acoustic cues of discourse such as pause, voice pitch and speaker identification. The system offers the user several levels of skimming. The first level plays back the recording at the normal pace. In the second level, audio is segmented by speech detection. Speech pauses (silence) under a certain duration threshold are removed while longer ones are reduced to a set value (the duration threshold). The next level attempts to take advantage of the natural structure and properties of discourse. Level 3 identifies salient points in the recording by considering longer pauses as *juncture* pauses, which would tend to indicate either a new topic, or a new speech turn. Hence, the system will only play back (for a certain amount of time) segments of speech which occurred after juncture pauses before jumping to the next one. Level 4, uses emphasis detection to identify salient segments of the recording. The emphasis detection algorithm is based on the speaker's pitch, or voice fundamental frequency (F0). An adaptive threshold for each speaker is generated which identifies points of highest pitch frames within the recording. The system will then only play sentences containing these highest pitch frames. These segmentation techniques are error-prone and will occasionally miss desired boundaries while mistakenly identifying others. The system compensates for these shortcomings by providing the user with additional navigation tools. These include a skimming backward mechanism which plays back the audio normally but jumps to the previous segment. This functionality enables a user who has heard something of

interest to pinpoint it's precise location. The user can also jump forward to the next segment if he decides that the current one is not relevant. Skimming audio through the highest levels of the system may be disorientating, as unrelated speech segments are played in fast successive order. A usability study of the SpeechSkimmer showed that users used the system at the highest skimming levels to navigate through the audio in order to identify general topic locations and then used lower skimming levels (normal play-back or pause compression-suppression) to listen to specific parts of the recording.

2.1.3 Automatic Speech Recognition

The field of Automatic Speech Recognition (ASR) has made significant progress in the last decade, evolving from single speaker, discrete dictation systems with limited vocabulary for restricted domains to sophisticated systems that tackle speaker independent, large vocabulary continuous speech recognition (LVCSR) tasks. Unconstrained LVCSR is a difficult tasks for a number of reasons including speech disfluencies in spontaneous dialogues, lack of word or sentence boundaries, poor recording conditions, crosstalk, inappropriate language models, out-of-vocabulary items and variations in accent and pronunciation. These conditions combined can cause substantial decreases in recognition rates (Furui 1999).

Speech Recognition is the task of automatically identifying a sequence of spoken words according to the speech signal (Rabiner and Juang 1993, Young 1995, Jurafsky and Martin 2000). In other words, given a sequence of word utterances w, recognition consists of finding the most likely word sequence $\widehat{w}$ given the observed acoustic signal S:

Equation 2.1

$$\widehat{W} = argmax_{\ W \in L}\ P(W|S) = \alpha.argmax_{\ W \in L}\ P(S|W).P(W)^{1}$$

A speech recognition process encompasses a number of successive steps based on a property of languages: the use of a limited amount of phonemes (the smallest perceptual "building blocks" of words), typically identified from between 40 to 60 distinct phones (basic sounds). Phones can be modelled using a Hidden Markov Model (HMM) containing a number of states and connected by transition arcs. These models can be combined together to form word models which in turn can be combined into sentence models. The first step of the recognition task will process the audio signal and extract a number of acoustic features over a certain time frame duration (typically 10ms). Features are chosen for extraction according to their ability to discriminate between different phones.

[1] *Using Bayes' Rule, where* $\alpha = \frac{1}{P(S)}$, $P(S)$ *being the probability of the acoustic observation sequence and is constant across all sentences,* $P(S|W)$ *the observation likelihood, or probability of observing a certain set of acoustic features given a sequence of words, is computed using the acoustic model and P(W), the word prior probability, or probability of observing a word independently of the speech signal, is computed using the language model.* $argmax_{\ X}$ *(f(x)) is the function which returns X such as X maximizes f(x).*

The observed acoustic features are subsequently translated into phone probabilities according to an acoustic model. The acoustic model consists of a pronunciation lexicon, where phones are usually divided into three states: beginning, middle and end. The triphone model further adds context to these states whereby individual phones are influenced by the surrounding ones. The *decoding* stage outputs the most likely sequence of words according to the word pronunciation dictionary and a language model. The language model assigns words prior probabilities according to some grammar inferred from a large corpus. A grammar defines allowable sequence of words and their probabilities. An example of such grammar is the n-gram model, where the presence of a word is deemed to depend only on the $n-1$ previous words. Probabilities of n-grams are thus computed by counting the number of occurrences of n successive words instances in a training corpus (word frequency for unigram model, word pair frequency for bigram model etc.) As the underlying language model explicitly models inter-word relationships, a misrecognition will often lead to another.

Although LVCSR has produced very encouraging results for certain task specific applications, serious challenges remain in recognising speaker independent spontaneous speech in unconstrained domains. Current research issues focus on building robust recognition systems by using automatic adaptation techniques, such as adaptation of acoustic models to speakers' voices and speech rate fluctuations, language model adaptation and improved spontaneous speech modelling (Furui 2003). Despite the above mentioned shortcomings, ASR is a central component to many audio browsing systems. Typically, the ASR module is used to produce conversation transcripts for convenient user scanning, reading and other text-based information retrieval operations.

2.1.4 Word Spotting

A keyword based retrieval query offers an alternative paradigm to full LVCSR transcription. Word Spotting consists of detecting the presence of a specific word or phrase in a speech corpus. This task is thus computationally far less expensive then generating full transcripts and may also be more appropriate for certain types of applications, such as querying a large audio database. Two types of errors can occur with a word spotting system, a *miss* and a *false-alarm*. A miss consists of not retrieving a particular keyword and a false-alarm of wrongly recognising one. Tuning a system requires finding an acceptable trade-off between correct keyword detection (true-hit) and false-alarm rates. The Receiver Operating Characteristic (ROC) is defined as the percentage of keyword detection as a function of false-alarm rates (in *fa/kw-hr*: false-alarm per keyword per hour). A Figure-of-Merit (FOM) is calculated as the average value of the ROC curve between 0 to 10 fa/kw-hr.

The application of an HMM-based recognition system to keyword spotting will typically require building acoustic and language models for a predefined set of keywords and non-keywords, or *fillers* (Rohlicek et al. 1989). Spotting a keyword then consists of two phases: hypothesising when a keyword may occur in speech (*putative hit*) and subsequently assigning a score to the hypothesis.

The hypothesis is accepted if the keyword score is above a rejection threshold. Thus, the output of a wordspotter would be a set of keywords and their time offsets, with everything in between considered as background words. Filler modelling is used to match arbitrary non-keywords present in speech and is crucial in the performance of the word-spotter. Appropriate models will reduce the rate of false-alarms, as shown by the comparative studies of filler models in Rose and Paul (1990). Another decisive component in the performance of the word-spotter is an appropriate scoring algorithm. DECIPHER (Weintraub 1993) assigns a likelihood score to an hypothesised keyword by combining acoustic likelihood probability and language model probability, where the language model is trained from combining task specific data (with high occurrences of the specific keywords) and task independent data. The main disadvantages of LVCSR based systems for word spotting is that they are computationally expensive and can only effectively recognise keywords if these are present in their lexicons.

To circumvent some of these shortcomings, an alternative approach to word spotting is off-line speech pre-processing to generate a phone lattice representation. The lattice representation consists of an output of multiple phone hypotheses at every speech frame, along with a likelihood score for the hypothesis (James and Young 1994). The depth of the lattice can hence be set by preserving only the n best hypotheses. Thus, word-spotting reverts to a keyword pronunciation match against each lattice. The main advantages of the phone lattice representation are that search is fast and that there is no restrictions on keywords. In Dharanipragada and Roukos (2002), speech is initially converted off-line into a table of phone trigrams with acoustic scores. This is followed by a two-step search, using the keyword phonetic transcription. If the query term does not appear in a pronunciation dictionary, a spelling-to-sound database generates the likely phonetic representation of the word. The first step is a fast coarse match which identifies keyword locations according to the phone trigrams index. In order to reduce the number of false alarms, this is followed by a detailed acoustic match.

2.1.5 Topic Segmentation

Automatic topic segmentation is the process of segmenting a (text or audio) document into regions of semantic relatedness. This is a difficult task for a number of reasons. First of all, as an abstract concept, a topic is difficult to define. Also, it is a subjective notion and topical annotation of documents by humans will often differ from annotator to annotator, particularly in the case of topic shifts. This is evidenced in Hearst (1994) where seven readers who were asked to find topical boundaries of a text document exposed a variety of judgements. Research on Topic Detection and Tracking (TDT) was originally targeted at newswire and news broadcast and typically involves three distinct phases. The first is to segment data streams into self-contained coherent units. A second phase consists in detecting new (previously unseen) topics. This step can either be performed on-line (as the news are broadcast live) or retrospectively, on a corpus of samples. The

final step consists of identifying whether incoming samples are related to a particular (target) topic. In the particular context of audio streams, all these operations are ideally performed using ASR transcripts for full automation. Therefore, the first audio segmentation step can be seen as a text segmentation task. The Dragon system (Yamron et al. 1997) requires a topic model for the segmentation task. A topic is modelled with unigram statistics. A training set is clustered into different topics using a distance metric. If a sample's distance to a given cluster is less than a certain threshold, the sample is included into the cluster and the cluster model is updated. If the distance is above the threshold, a new cluster is created. Once the topic model has been created, segmenting a stream is done by scoring stream frames against the topic model and detecting topic transitions. Another approach to segmentation, described in Allan et al. (1998), measures shifts in the vocabulary. Each sentence of a text stream is run as a query against a local context analysis (LCA) thesaurus which identifies and returns a number of semantically related words or concepts. While some sentences of the original text have few or no common words, they may in fact share a number of concepts. The text is then indexed at the sentence level according to these features. A function of the feature offsets is then used as a heuristic measure of change in content. The chief advantage of this approach is that it is unsupervised. Its drawbacks are that it is computationally costly (a database query per sentence) and that LCA results for sentences with poor semantic value (a common feature of speech) are essentially random. Another approach consists in modelling lexical features or "marker words" usually found at the start and end of topical segments in order to predict topic changes.

The approaches mentioned above make exclusive use of textual features while ignoring some of the specific characteristics of speech such as *prosody*. Prosody (in linguistics) refers to phonological features in speech such as syllable length, intonation, stress, and juncture, which convey structural and semantic information. In addition to lexical information obtained from speech recognition, Tur et al. (2001) use prosodic features automatically extracted from speech for automatic topic segmentation. A distinctive advantage of using a prosodic model is that it is largely independent of the recognition task and therefore should be robust to recognition errors. The topic segmentation algorithm is implemented in two phases: the speech input is first segmented into sentences (speech units). Then sentence boundaries are analysed to determine whether they coincide with a topical change. In effect, this approach reduces topic segmentation to a boundary classification problem, i.e. estimating the probability of a topic boundary given a word sequence and its set of prosodic features. To this end, a prosodic model needs to be created, built on the feasible extraction of prosodic features for a fully automated solution. A corpus with human labelled topic boundaries was used in order to infer useful prosodic features. Features which were found to be important in identifying topic boundaries include: pause duration at boundaries, pitch or fundamental frequency across boundary, last phone duration before boundary (Shriberg et al. 2000). In addition, non-prosodic features which were available from the speech recogniser, such as speech turns and speaker

gender were also included in the model. The prosodic model was subsequently combined with a language model similar to the one used in Yamron et al. (1997). The performance obtained was comparable to that of the best word-based systems.

2.1.6 Spoken Language Summarisation

Unlike automatic text summarisation, which has long been a subject of study, spoken language summarisation is a new research domain, with serious issues remaining to be solved. These include how to deal with the presence of speech disfluencies in spoken dialogue, sentence boundaries, information spanning across several speakers and speech recognition. Speech disfluencies include non-lexicalized filled pauses (*um, uh*), lexicalized filled pauses (*like*), repetitions, substitutions and false starts. DiaSumm (Zechner and Waibel 2000) is a spoken language summarisation system comprising a number of stages. Audio recordings can theoretically be used as input. However, the results described bellow were obtained using human generated transcripts with annotated topic boundaries (Zechner 2001). The system first runs a part of speech (POS) tagger on the transcripts to identify disfluencies. Repetitions and discourse fillers are subsequently removed through a clean-up filter algorithm. The result of the POS tagger is then fed into the sentence boundary detection component. False starts are then detected and removed. Cross-speaker information detection consists of identifying question-answer pairs, which is first done by detecting questions and then the corresponding answer. Once all these steps are completed, the summarisation mechanisms rank sentences using a TF-IDF (Term Frequency, Inverse Document Frequency) based MMR (Maximum Marginal Relevance) ranking within topical segments. This algorithm is intended to extract salient parts of the document while avoiding redundancy.

Definition 2.1 TF-IDF

The TF-IDF (Term Frequency, Inverse Document Frequency) of a term t_k with respect to a document D_j in a set of documents T_r is given by

$$TF - IDF(t_k, D_j) = nt(t_k, D_j) \times \log \frac{|T_r|}{nd(t_k, T_r)}$$

where $nt(t_k, D_j)$ is the number of times t_k appears in D_j and $nd(t_k, T_r)$ is the number documents from set T_r with at least one occurrence of t_k.

TF-IDF reflects the intuition that the more a term occurs in a document, the more it is representative of that document, and that the more a term occurs across various documents, the less discriminative it is. MMR (Carbonell and Goldstein 1998) rewards "novelty" by allocating increased weight to a document if it is: (i) both relevant to the query and (ii) has little similarity with previous selected documents.

Valenza et al. (1999) present a speech summarisation system which combines inverse term frequency with an audio confidence measure from the speech recogniser's output. For a word to be included in a summary it needs to have high probabilities of relevance and correct recognition. The authors stress that in order to produce useful summaries, a certain level of inaccuracy should be acceptable. Giving too much weight to audio confidence risks omitting relevant information from the final summary. The acoustic confidence measure for a particular word is determined by the sum of phone probabilities for that word normalised by word duration. Summaries are generated on a per-minute basis to favour spread content over punctual information and may be more adapted to the targeted audio (broadcast news). A summary can be a set of keywords (frequently occurring single words), n-grams (n words extracted from the audio transcript, with n determined by the user) or utterances (audio segments delimited by speaker or content change). The user interface provides a keyword list, a user-specified summary type as well as full text output. Selecting a keyword causes relevant segments of the summary and full text to be highlighted. The user can also listen to the corresponding audio segment or to larger audio segments. The system thus provides audio indexing and summarisation.

An approach which unlike the above does not rely on lexical recognition was introduced in Chen and Withgott (1992). It uses pitch and energy content to detect emphasis and create summaries by selecting emphasized segments in temporal proximity.

2.2 Video Browsing

A video document essentially consists of a succession of images over time, but will often contain additional modalities such as sound and text. Therefore, indexing a video document can be approached from the various modalities it may contain. Since we have covered techniques for audio document browsing in detail in the previous section, here we will describe visual and multimodal approaches to video indexing from a meeting recording point of view. Current approaches to video browsing for the most part employ techniques which rely on domain knowledge of video types and thus make a number of assumptions about features of the recording which limit their applicability to meeting browsing. The availability of closed-captions in news broadcasts, for example, may be used to generate summaries using text-based techniques. Sports video indexing techniques and highlight extraction generally use heuristics valid only in the context of the rules, grammar and semantics of a specific sport (though more generic approaches to sports video indexing have also been investigated by Hanjalic (2003)). High motion, high pitch and increased audio volume may be used to identify action scenes in feature films but would be of limited use in recordings of typical meetings. In Snoek and Worring (2005), techniques are reviewed which regard video from an author's perspective, assuming a process of production and editing which defines documents with clear structure and semantics, where scenes and transitions can be identified. In Smith and

Kanade (1998), selection of static frames preceded by scenes of camera motion, or zooming are among a number of heuristics used to choose frames of importance. Such assumptions, while valid in a production environment, are mostly inadequate in the case of automatic meeting recordings, which typically contain raw data captured from a number of unmanned fixed cameras. In Li et al. (2000), a survey of browsing behaviour for various types of video content concludes that as information in conferences is essentially audio centric, visual features only offer users minimal cues on content. In what follows and unless otherwise stated, we present a number of techniques suitable for indexing, segmenting and browsing meeting recordings captured by unmanned static cameras in a conference room.

2.2.1 Visual Indexing

A scene or *shot* can be defined as a succession of images which have been continuously filmed and constitutes an intuitive fundamental unit in video. One common technique for automatic video segmentation is automatic *shot boundary detection* which is achieved through the non-trivial task of measuring similarity (or rather dissimilarity) between successive frames over a certain number of features of the image (colour, texture, shapes, spatial features, motion etc.) A number of reliable methods have been proposed to this end (Aigrain et al. 1996). When a video document is created through a production process, changes between shots may not necessarily be clear cut but use a fading, dissolving or wiping transition. In order to detect gradual transitions, algorithms for boundary detection generally include a dissimilarity accumulation function with a boundary found if the functions goes over a certain threshold. Once a video has been segmented into shots, these can be characterised by a single image, chosen according to certain heuristics (e.g. choose first frame, frame containing a face or significant object). Time-varying spatial information can thus be translated into the spatial domain for convenient scanning through the use of *keyframes*, whereby each scene of a video can be represented with a single image, offering a visual summary of the recording. Although these methods can be valuable for feature films or video database indexing, their application to meeting recordings is certainly limited as significant visual changes in a common meeting scenario are likely to be minor (e.g. drawing on the board), thus their discriminating power weak and their semantics rather limited without additional (audio) information.

Another promising research area in automatic meeting segmentation and indexing is concerned with identifying specific and significant meeting *actions*. McCowan et al. (2005) propose using low-level audio and visual features (speech activity, energy, pitch and speech rate, face and hand blob) to model meetings as a continuous sequence of high-level meeting group actions (monologue, presentation, discussion, etc) using a HMM model based on the interactions of individual participants.

2.2.2 Video Summarisation and Skimming

Similar to the familiar fast-forward feature of standard video players, time-compression can be used to increase the speed of image play-back in digital video recordings. However, speed increase is inversely proportional to a viewer's ability to understand the recording and this technique will quickly result in a serious degradation of comprehension. It is also cumbersome for browsing lengthy recordings with no specific reference points. Another technique consists of displaying frames separated by a fixed time interval (e.g. 30s) but this process is essentially random, might skip over crucial information, is generally confusing and remains time consuming. In the study of video browsing behaviour carried out in Li et al. (2000), users navigated conference presentation recordings using essentially time compression and speech-based silence removal techniques. An interesting alternative to these fast-forwarding techniques is used in CueVideo (Srinivasan et al. 1999). Sequences with low motion are sampled and played with fewer frames thus faster than ones with higher motion levels in order to quickly skip over scenes with little content information and jump to significant ones. This technique, also known as variable speed fast-forward, seems particularly well adapted to meeting recordings as they often contain long shots with little or no significant motion. The drawback of this technique is that faster video sequences cannot be synchronised with audio in an intelligible way. It remains however an efficient tool for navigation. The Informedia^{TM} (Smith and Kanade 1998) project at Carnegie Mellon University offers search and retrieval of video documents from an online digital library. The system integrates image analysis and speech and language processing techniques to produce *skims* of video documents. Keywords are identified through audio transcripts and closed captions (where available) using TF-IDF. A compressed audio track is then generated according to the location and duration of these keywords. The number of keywords retrieved therefore defines the duration of the skim. Once the audio track has been created, a corresponding video skim is generated. To avoid redundancy within close proximity, a keyword cannot be selected twice within a certain number of frames. A minimum (2s) of matching video frames is played along with the keywords from the audio track for clarity, but the video segments are not necessarily time-aligned with the audio. Alternative frames of a corresponding scene are picked according to certain heuristics. These include prioritising introduction scenes, frames with human faces, static frames preceded by camera motion, zoom etc. Compaction ratio is typically 10:1 but a ratio as high as 20:1 has been found to preserve essential information.

We have alluded to the fact that many domain knowledge assumptions used in broadcast news, feature films and sports video browsing may not fare well when used with unstructured meeting recordings. However, regular meetings in a given organisation may also follow a well structured grammar, which can then be exploited for meeting summarisation. VidSum (Russell 2000) uses regular patterns occurring in weekly staff forums (e.g. introduction by first speaker, presentation by second speaker, applause, questions and answers) to generate concise summaries of presentation

recordings. Content analysis first extracts a number of visual features, which are then matched against a presentation library in order to find the most likely presentation structure. The next step is to populate a video template called *summary design pattern* (SDP). Slots in the SDP are filled according to priority criteria (e.g. introduction, conclusion) until a pre-determinate structure and time constraints are met. The result is a concise, well-structured and pleasant to watch summary. However, evaluating the summarisation process remains difficult since, as remarked in Russell (2000), different templates may produce significantly different summaries.

2.3 Artefact Browsing

This study is essentially concerned with reviewing automated solutions for accessing meeting recordings, therefore we are primarily interested in tools and techniques which require no additional effort from the participants during the actual meetings. However, there is another important category of meeting access tools, which we will refer to as *meeting minutes systems*. Minutes systems provide support for note taking, meeting annotation and thus for later access to meeting recordings through active effort by the participants *during* the meeting process. Although a comprehensive review of all meeting support tools is beyond the scope of this article, we will describe a number of meeting minutes systems and analyse some implications of note taking for browsing.

2.3.1 Meeting Minutes Systems

NoteLook (Chiu, Kapuskar, Reitmeier and Wilcox 1999) is client-server system deployed in a media-enriched meeting room to support multimedia note taking. Participants take notes during the meetings through the NoteLook client, which runs on wireless pen-based notebook computers. Presentation material displayed by a room projector, images of the whiteboard, video of the speaker standing at a presentation podium and room activity are some of the data participants can incorporate into their personal minutes, either as still images or video streams (recorded by the server). The users can select which live video channel is displayed on the client, and still images can be incorporated into the notes either as thumbnails or as the page's background image. For slide presentations, NoteLook provides an automatic note-taking option which captures any new slide transition and generates thumbnails of room activity at regular intervals during a slide duration. Images and pen strokes are time-stamped and can therefore be later used to access the video recordings of the meetings. LiteMinutes (Chiu et al. 2001) is an applet-based note-taking application running on a wireless personal computer. Meetings take place in a media-enriched conference room and video, audio and slide images are a number of potential multimedia items captured and stored at a server. Notes taken during the meeting are timestamped. They can be viewed in real-time by other meeting participants (if a designated person acts as a scribe) and can also be revised later on. Notes taken on different laptops are handled separately. After the

meeting, notes can be e-mailed to designated recipients and are also accessible through the capture server, which hyperlinks the notes to related media (slide, video) if these were active at the time of writing (smart-link). MinuteAid (Lee et al. 2004) is a meeting support system which enables participants to request and embed meeting multimedia items within a Word document during a meeting. Multimedia items which can be requested by the MinuteAid client running on a participant's personal computer include projected slides, audio recordings, omni-directional video and whiteboard images. Slides can be obtained in real-time, audio tracks require a 15 second delay while video can only be obtained once the meeting has ended and the video recording has been processed by the server. Once all data requests have been processed, participants can manipulate the minutes as a standard multimedia document.

2.3.2 Implications for Browsing

In most meeting scenarios, participants will interact with artefacts of some sort, to present and share information (slides), express and clarify ideas (whiteboard) or as personal minutes (note taking). Thus, actions associated with artefacts will generally be associated with significant meeting events and will convey strong semantic content. A number of researchers have investigated participants' interactions with meeting artefacts as a means of segmenting, indexing and structuring meetings. Filochat (Whittaker et al. 1994) is a digital notebook which enables audio indexing of collocated meetings through note taking. Time indexed handwritten notes allow users to listen to concurrent segments of audio. An important and unforeseen result of a usability study of the device was that some users made explicit indexing notes during meetings when hearing subjects of potential interest in order to revisit these specific points later on. Audio indexing according to note-taking activity is also implemented in the Audio Notebook (Stifelman et al. 2001), a paper note-book with cordless pen coupled with a digital audio recorder. Audio indexing is complemented by speech skimming functionalities through speed control of audio play back, phrase detection (which prevents audio from being played from the middle of a sentence) on topic shift detection, based on acoustic features of the audio recording (pitch, pauses and energy). Users of the Audio Notebook were able to use the functionalities provided by the system to successfully review recorded information, clarify ambiguous or misunderstood notes, and retrieve portions of audio which had been intentionally bookmarked. Classroom 2000 (Brotherton et al. 1998) is an educational system which aims to give students access to the content of university lectures. The system provides access to audio and video recordings as well as additional information such as web documents visited during the lecture and note written on an electronic whiteboard. There are several levels of access to a lecture: slide transitions, which provide access to the audio for a duration of each slide, pen-stroke level, which provides access to audio for the writing duration and word level. To facilitate navigation of recorded lectures, the system displays a time line indexed with all significant events captured during the lecture.

2.4 Meeting Browsers

Meeting browsers are systems that integrate some or all of the above described technologies in order to provide information seekers with a unified interface to multimedia meeting archives. We here present the most important meeting browsing systems and their user interface, and describe how they integrate various modalities for browsing meeting recordings.

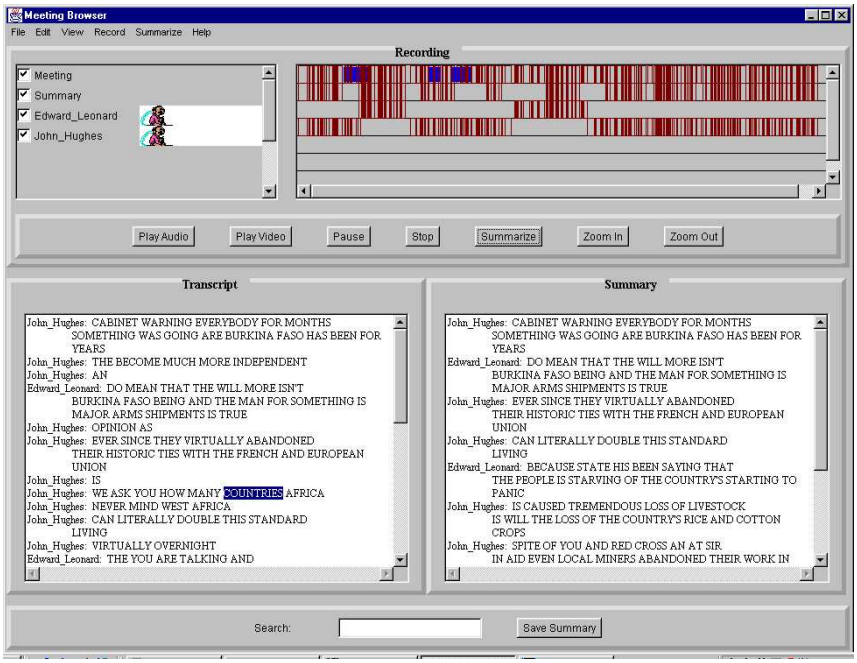


Figure 2.1: The Meeting Browser user interface

2.4.1 The Meeting Browser

The Meeting Browser (Waibel et al. 1998, 2001) displays meeting transcripts time aligned with corresponding sound or video files. The browser, seen in Figure 2.1, comprises a number of components, including a speech transcription engine and an automatic summariser. The summariser attempts to identify salient parts of the audio and present the result to the user as a condensed script, or *gist* of the meeting. The summariser takes a textual transcript as an input. This transcript is either generated manually or from a speech recognition run. The summarisation algorithm works as follows: identify the most common stems present in the transcript and then weight all speech turns accordingly. The turns with the highest weights are then included in the summary. These most commons stems are then removed and the process is repeated over turns not previously included until the summary has reached a predefined size. Several experiments were designed in order to evaluate the summarisation system. The first one involved asking users to categorize 30 dialogues into a certain number of predefined categories according to a ten turn summary of the dialogues. The authors report a precision of 92.8%. Another task was to ask users to answer a

number of questions based on a summary of the dialogue. The dialogue transcript used in this case for summarisation was generated by speech recognition. The user could decide (and increase) the number of turns included in the summary. With the number of correct answers increasing with the number of speech turns included, the authors claimed this demonstrated the potential use of speech recognition output for summarisation while conveying important points of a dialogue.

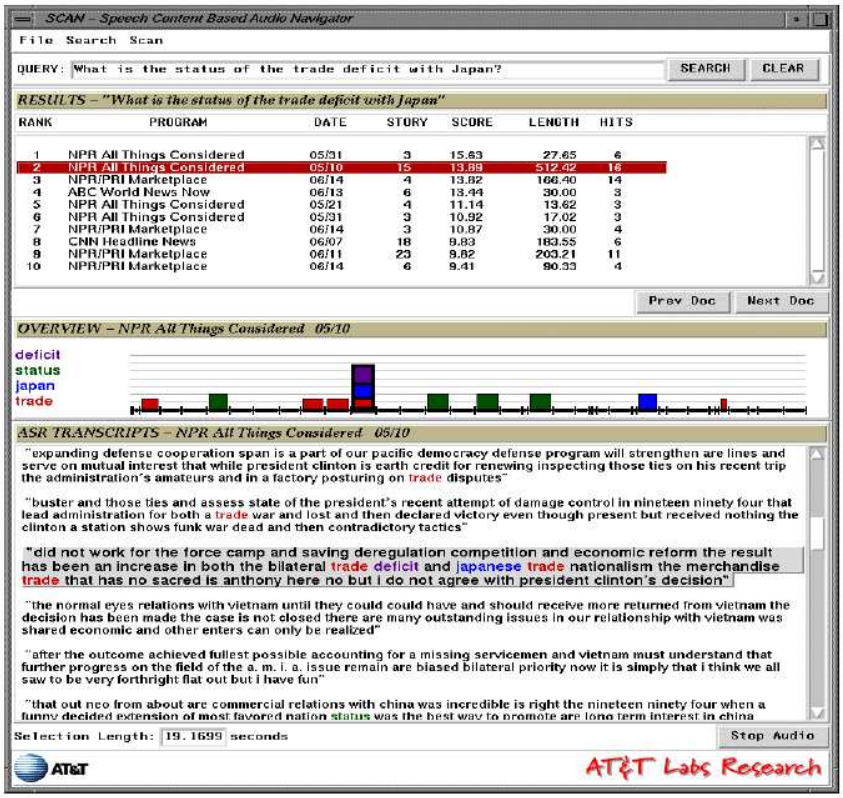


Figure 2.2: The SCAN user interface

2.4.2 The SCAN system

SCAN (Spoken Content-based Audio Navigation) is "a system for retrieving and browsing speech documents from large audio corpora" (Choi et al. 1999, Whittaker et al. 1999). SCAN uses machine learning techniques over acoustic and prosodic features of 20 ms long audio segments to automatically detect intonational phrase boundaries. Intonational phrases are subsequently merged into intonational paragraphs, or *paratones*. The result of the intonational phrases segmentation is then fed into a speech recognizer (around 30% word-error rate) which uses the automatic transcripts generated for a document retrieval system based on the vector space model of weighted terms. SCAN introduces several interesting mechanisms as an information retrieval system. The first one is called *query expansion* which adds related words (located within high-ranking documents) to users' short queries. The second one is called *document expansion* and attempts to compensate for some of the errors due to speech recognition. It uses the best recognition output for a given

audio document as a query on the audio database. The top 25% of words present in the original document word lattice (and not included in the final transcript) and in at least half of the highest ranking documents retrieved by the query are subsequently added to the original document transcript. Both techniques improved the information retrieval tasks. SCAN's user interface, seen in Figure 2.2, has three components: search, overview and transcript. The search component retrieves audio documents based on users' queries match against the ASR transcripts of the documents contained in the database. The ten highest ranking documents are displayed along with the number of hits (number of terms of the query contained in the document transcript). The overview displays the audio document segmented along the paratones mentioned above, with their width proportional to their duration. Terms from the user query contained in the speech segments are represented by a colour coded rectangle, whose height is proportional to term frequency. The transcript view displays the paratones' ASR transcripts. Clicking on them will play the corresponding audio segment.

2.4.3 Video Manga

Video Manga (Uchihashi et al. 1999, Boreczky et al. 2000) is a video system which automaticaly creates pictorial summaries of video recordings. The system was primarily tested and evaluated on recordings of collocated meetings in a conference room but it was found to also work well with other video genres (films, commercials). Although recordings were not edited after the meetings, an operator was in control of meeting capture and could pan and zoom as well as switch between a number of cameras and other displays. This would naturally tend to encourage the capture of highlights, which would not occur with unmanned fixed cameras. Video Manga generates summaries of a meeting as a chronologically ordered compact set of still images, similar to a comic strip, hence the name. Clicking on a specific keyframe will play the corresponding video segment. The keyframe extraction technique does not simply rely on shot boundary detection but on a colour histogram based hierarchical clustering technique which identifies groups of similar frames, regardless of timing. Once video segments have been identified, an importance metric is used to reward segments if they are both long (a heuristic suited to the specific manned capture environment) and unusual. Segments which score less than one eighth (empirical threshold) of the maximum scoring segment are discarded (another option is to precisely select the number of segments included in the summary). For meeting recordings, this threshold led to discarding around 75% of the frames. In order to give higher visual importance to better scoring segments, a keyframe size in the final summary varies on a scale from one to three according to importance score. The selected frames are further reduced by removing consecutive frames from the same clusters and similar frames which are separated by only one single frame from another cluster (e.g. in dialogues). The frames importance score can also be weighted according to human, groups, slides shots detection. Documents such as slides, web pages, transparencies displayed in the meeting room

are captured every five seconds. The text from these documents is timestamped and can be used to label corresponding shots in the pictorial summary.

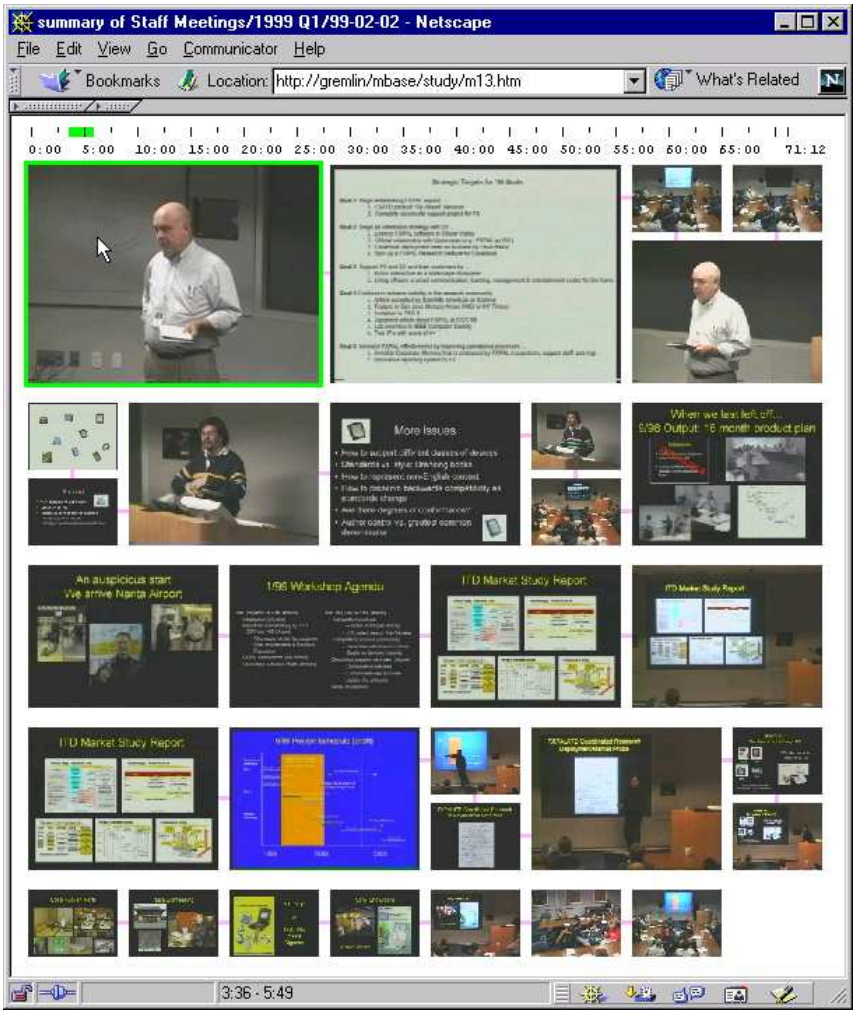


Figure 2.3: A meeting summary produced by Video Manga

2.4.4 The Portable Meeting Recorder — MuVie

The Portable Meeting Recorder (Lee et al. 2002, Erol et al. 2003) is a system that captures a panoramic view of meetings and detects speaker location in real-time. Post-meeting processing of the recorded data generates a video skim which focuses on participants according to speech activity. As there would normally be minimal motion during a typical meeting, the authors argue that segments of higher motion potentially indicate significant events such as a participant joining the meeting or doing a presentation. Similarly, segments of higher speech volumes may point to phases of intense discussions, particularly when coupled with information about speakers' locations (high number of exchanges). The MuVIE (Meeting VIEwer) user interface, seen in Figure 2.4, thus provides among other information about the meeting (keyframes, transcripts) a visual representation

over a timeline of audio and visual activities and speakers' turns. A meeting summary can also be generated by playing back in time order the video segments containing the highest visual or audio activity and highest ranking keywords extracted from the meeting transcripts.

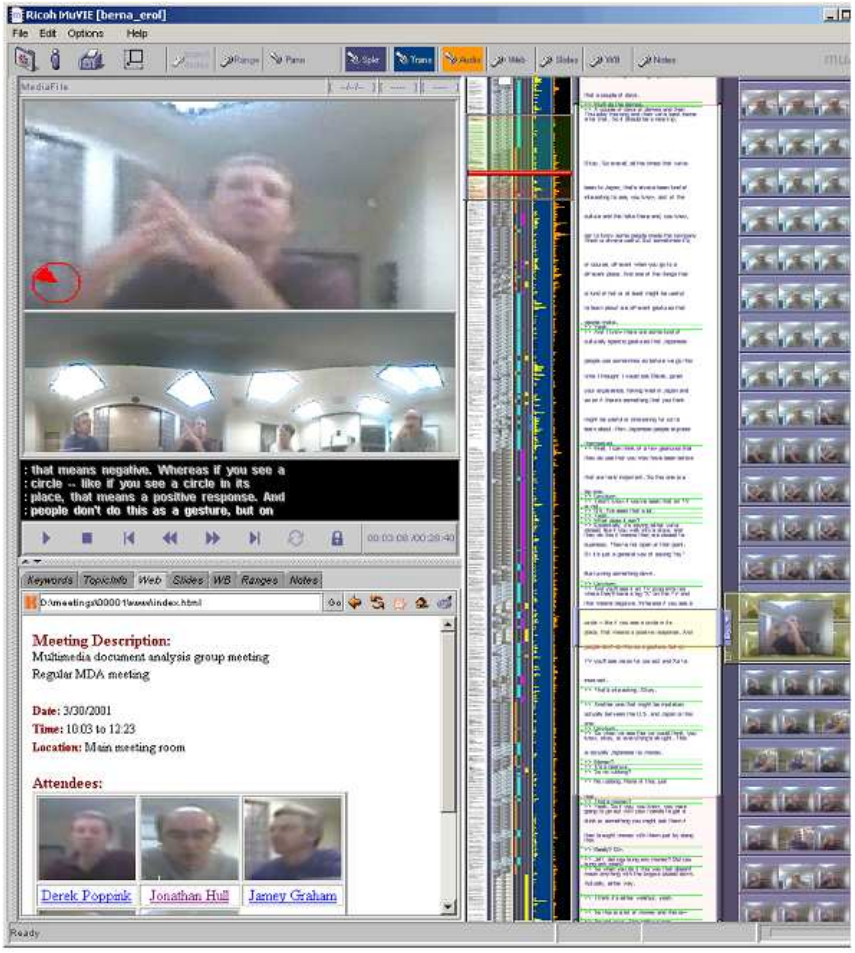


Figure 2.4: The Muvie client user interface

2.4.5 The MeetingViewer

The MeetingViewer (Geyer et al. 2003) is a client application for browsing meetings recorded with the TeamSpace (Geyer et al. 2001, Richter et al. 2001) online conferencing system. The TeamSpace client provides low-bandwidth video for awareness feedback and supports the use of a number of artefacts such as sharing and annotating slide presentations, creating and editing agenda and meeting action items and inserting bookmarks. In addition to session events (joining, leaving meeting) all interaction events performed on the client are automatically recorded and timestamped by the server. These events are subsequently used to index the meeting and are displayed on a timeline on the MeetingViewer interface to facilitate navigation (Figure 2.5). The user can thus choose relevant sections of the meeting. Playback will play corresponding segments

of the audio and video recording along with all concurrent meeting events. Specific artefacts may be picked for viewing through the use of a tabbed pane.

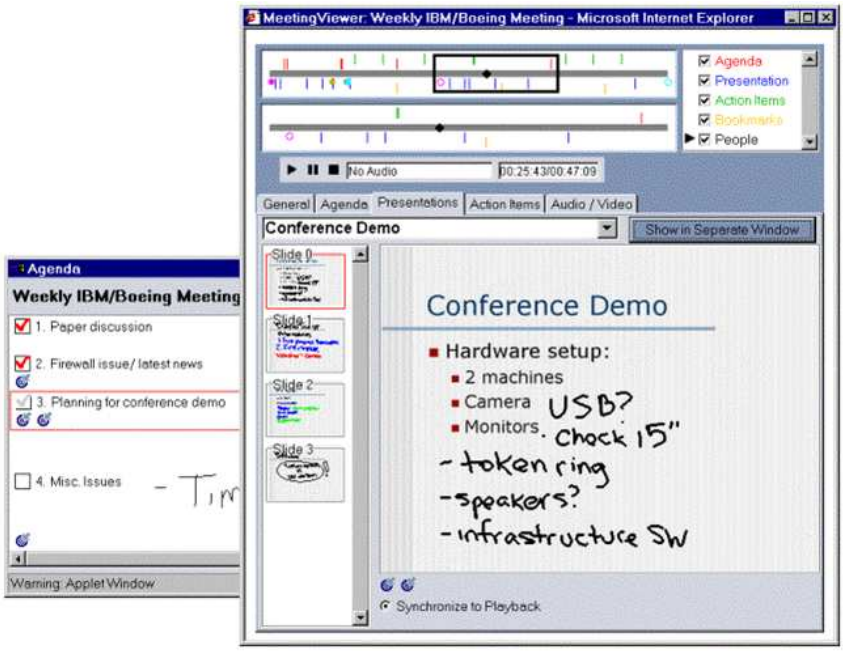
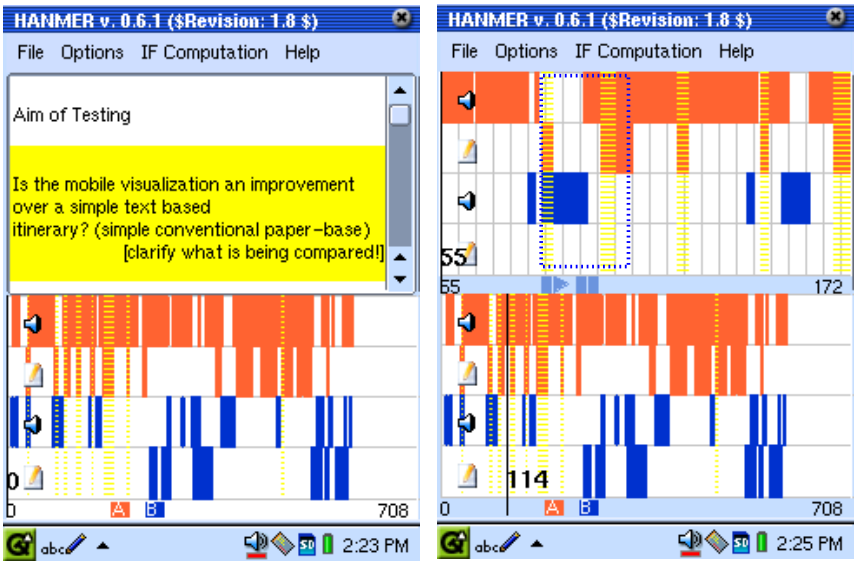


Figure 2.5: The MeetingViewer User Interface

2.4.6 COMAP and HANMER

COMAP (COntent MAPper) (Luz and Masoodian 2005) is a system for browsing captured online speech and text meetings using the concepts of *temporal neighbourhoods* and *contextual neighbourhoods*. These concepts are based on viewing meeting data as a collection of discrete events, or segments. A temporal neighbourhood is defined as concurrent media events as well as segments related to these events. Segments are in a contextual neighbourhood if they share some content features (keywords). The system takes as input an audio recording along with an XML file containing detailed metadata about participants' edits and gesturing (telepointing) actions. These action metadata are automatically generated by RECOLED (Masoodian et al. 2005), a shared-text editor designed for this purpose. The user interface displays the textual outcome of the co-authoring task along with mosaic timeline views of the participants' speech and editing activities. To browse a meeting, a user can click on a portion of text which will highlight audio segments in the temporal neighbourhood of that text segment. The user can listen to an audio segment by clicking on it, which in turn may highlight potential concurrent editing operations. An *Interleave factor* (IF) metric (Luz 2002) measures levels of concurrent media activity, with intervals of greater activity deemed to be of greatest significance. A summary view of a recording can be generated through IF ranking. HANMER (HANd held Meeting browsER) (Luz and Masoodian 2004) provides the same functionalities as COMAP but was designed for portable devices (Figure 2.6).



(a) (b)

Figure 2.6: Hanmer User Interface

2.4.7 The Ferret Media Browser

The Ferret Media Browser (Wellner et al. 2004) is a client-server application for browsing recorded collocated multimodal meetings. Recorded data include video, audio, slides (on computer projection screen) as well as whiteboard strokes and individual note-taking (digital pen-strokes), which are timestamped. A tabletop microphone array permits speaker identification. Upon starting the Ferret browser, the user can pick a combination of any available media for display and synchronised play-back. ASR transcripts, a key-word search and speech segmented according to speakers' identity are also available. The user can zoom on particular parts of the meeting. Media streams can be dynamically added to or removed from the display during the browsing task. Other data sources can also be accessed through the internet.

2.4.8 WorkspaceNavigator

WorkspaceNavigator (Ju et al. 2004) is designed to provide access to information on loosely structured collaborative design projects which lasted over a long period of time in a designated workplace. Unlike most of the other systems described here, the data recorded for meeting documentation do not include audio and video media streams but rather discrete events. This design choice is motivated by the fact that (i) given the long duration of the design process, recording live streams of all activities would produce a prohibitive amount of data and (ii) the assumption that still images are often sufficient to jog participants' memories. Information on the design process is captured implicitly but participants can also explicitly capture specific events should they wish to do so, for later reference. Implicit data capture is performed every thirty seconds and includes an overview image of the activity in the workplace, motion events, computer screenshots as well

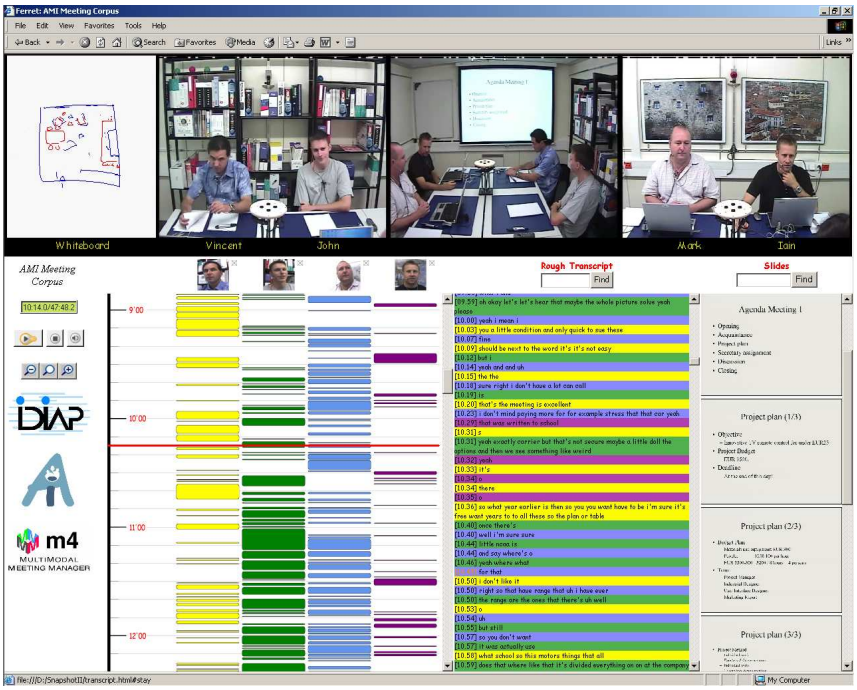


Figure 2.7: The Ferret Media Browser

as opened files and web resources and shots of operations performed on the whiteboard. In addition, participants can chose to capture the state of the whiteboard and integrate images and annotations to the project's documentation at any time. A number of usability studies performed on WorkspaceNavigator demonstrated the usefulness of implicit discrete information capture for design process documentation, data recovery and specific information item retrieval.

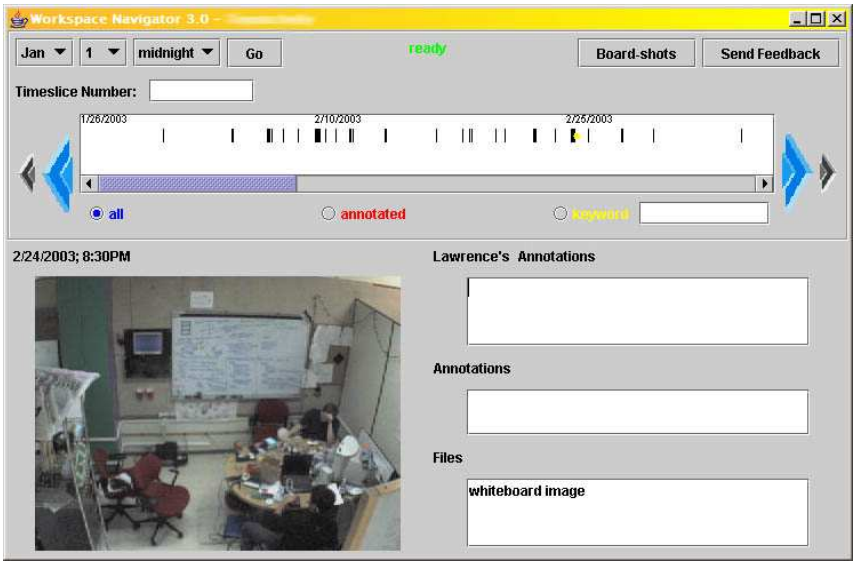


Figure 2.8: WorkspaceNavigator User Interface

2.5 Meeting Browser Evaluation

Meeting browser systems are notoriously hard to evaluate. Unlike speech recognition and spoken document retrieval, for which the TRECs 6-8 (Text REtrieval Conference) tracks (Garofolo et al. 1999) set precise evaluation tasks, with specific evaluation metrics on well defined corpus collections, the diversity of multimodal meeting recordings and browsing strategies makes defining evaluation metrics and system comparison impractical. System comparisons have been confined to assessments of information retrieval performance on multimodal meeting browsers against a baseline system, typically one based on a tape-recorder interface metaphor, as in Whittaker et al. (1999). A more common, less constrained but inherently less comparable approach is evaluation by usability testing. Tasks employed in usability testing have included using the browser for identifying the topic of a conversation (Zechner and Waibel 2000), classifying media items into predefined categories (Waibel et al. 1998), answering specific questions about meetings (quiz) (Erol et al. 2003), locating specific information items (Boreczky et al. 2000), and producing meeting summaries. Evaluation focuses on user feedback such as ranking of features of the user interface according to perceived usability (Boreczky et al. 2000) and overall impressions of system performance (Valenza et al. 1999) which give a good indication of how fit a system is for general use. In Uchihashi et al. (1999), manual minutes generated by a scribe during the meetings are used as a benchmark. Automatically generated meeting summaries were analysed to quantify the number of significant events they were able to convey. Information contained in the minutes which could not be inferred from the complete meeting recording (e.g. information external to the meeting or outside camera range) was not taken into account in the performance measure as it could not possibly have been included in the video summary.

Recently, more systematic approaches to comparing meeting browser performance have started to emerge. A strategy is described in Wellner et al. (2005) which suggests using the number of *observations of interest* uncovered by system users in a certain amount of time as an evaluation metric. A test (Browser Evaluation Test or BET) is proposed which can be described as follows. Human observers review information of interest in recorded meetings. Test subjects are then asked to answer as many (true-false) questions as they can in a period corresponding to half the duration of the meeting. Although this methodology is general enough to be used with most meeting browsers, it alone does not solve issues relating to the diversity of corpora and access modalities, and therefore does not suffice for performance comparison. Standard meeting corpora, such as the AMI meeting corpus (Carletta et al. 2005, AMI Corpus 2006) have become available which might help alleviate this problem. A crucial issue relating to usefulness and structure in browsing tasks is that of how to locate and "salvage" (recover with the purpose of creating a summary of the meeting) observations of interest. An interesting study on this issue is presented in Moran et al. (1997). Although that investigation is set in the context of designing meeting capture support tools, it offers valuable insight into how meeting browsers can be evaluated.

2.6 Summary

We have presented an overview of existing methods for segmentation, indexing and searching of captured multimedia meeting data and introduced a number of browsing systems, underlining their individual approaches to integrating various modalities for navigation of meeting recordings.

Due to the nature of the information contained in meetings, mainly audio-centric, most systems presented in this chapter essentially rely on browsing techniques based on various representations of participants' speech, such as speaker segmentation, transcripts and speech summaries. These are usually complemented by additional representations (screenshots, keywords)when other media sources are available from the meetings (video, meeting minutes, etc.) We concluded by highlighting the difficulties in evaluating meeting browsing systems and described a number of techniques used for this purpose. We briefly introduced a more systematic approach (BET)proposed by Wellner et al. (2005). We will use this evaluation test in our own user-oriented evaluation described in Chapter 9 and extend it with additional performance metrics.

While the next chapter describes background multimedia modelling concepts, the rest of this thesis will describe models, systems implementations and evaluation techniques which relies less on traditional speech representations but focus on participants' non-verbal interactions in order to support meeting browsing.

Chapter 3

Temporal, Specification and Content Models in Multimedia

In the previous chapter, we reviewed the state-of-the-art of meeting browsing, presenting the various technologies which are currently used for segmenting, indexing and retrieving information from multimedia meeting recordings. We have presented how various researchers have integrated some of these techniques, depending on their own interests or the meeting corpora at their disposal, to produce a number of meeting browsing tools. An important area which is currently lacking within the meeting browsing research community is the near absence of formal models. It is therefore one of the ambitions of this thesis to contribute to the state-of-the-art of meeting browsing by proposing a formal model for meeting information capture and retrieval, based on participants' interactions during meetings. This model relies heavily on existing concepts and formalisms of multimedia models, and due to the nature of interaction information, particularly on *temporal* models.

Temporal models, in computer science, were initially essentially concerned with temporal *specification* models: how to describe the temporal behaviour of a multimedia document so it could be correctly represented at play-back (i.e. behaved as intended by the designers of the multimedia document). More recently, as a large number of technical issues relating to multimedia synchronisation have been satisfactorily resolved, members of the research community have started to switch their focus on creating higher level multimedia *content* description models (e.g. MPEG-7), with the main goal of subsequently facilitating retrieval of information in ever increasing volumes of multimedia data. In this chapter, we will briefly review a number of multimedia modelling concepts which will be used recurrently through the rest of this thesis.

3.1 Overview

A multimedia document is by definition composed of heterogeneous data types (e.g. text vs images) and will also often contain a mixture of static data (e.g. text, graphics) and time dependent data (e.g. audio, video) which properties vary over time. Due to the extremely large variety and possible goals of multimedia applications, an equally large number of models have been formulated by the research community over the years for the representation of multimedia data. In order to provide ordering and coordination between various and possibly heterogeneous media elements, synchronisation models had to be developed in order to specify time constraints between elements, with application-specific synchronisation mechanisms responsible for enforcing these timing constraints at runtime. Some of the models were specifically created to answer the needs of a particular application. However, efforts were also made to formulate more generic models which could be equally applied to a wide range of multimedia data and systems. A number of multimedia composition and temporal specification models for multimedia documents presentation and play-back have been described in state-of-the-art reviews such as the ones carried out by Blakowski and Steinmetz (1996) and Bertino and Ferrari (1998). While earlier multimedia models were essentially concerned with issues relating to multimedia synchronisation, recent efforts have seen the emergence of standards such as MPEG-7, which aim at adding descriptive content annotation and structure to multimedia data. In this chapter, we will review some of the generic temporal models which have been proposed over the years to describe the semantics of temporal relations of time-dependant multimedia data. We will then take a closer look at a number of synchronisation specification models, while ignoring the details behind application-specific synchronisation mechanisms. Finally, we will introduce emerging multimedia content description models such as MPEG-7.

3.2 Multimedia Temporal Models

3.2.1 Multimedia Synchronisation Issues

In multimedia data, temporal relations between various elements can be inherent, such as the ones which implicitly existed at the time of capture (e.g. audio and video in a live recording). This is also known as *natural* synchronisation. Timing constraints can also be arbitrarily specified by the author of a multimedia document (e.g. CD-ROM slide presentation with corresponding audio). This is known as *synthetic* synchronisation. To ensure that the timing constraints between various time-dependant and time-independent multimedia items are preserved, synchronisation models have been developed to ensure timely delivery of multimedia objects, within the requirements of a specified (or acceptable) Quality of Service (QoS).

Intra media synchronisation ensures that a single time-dependant media (e.g. speech) is played back at the appropriate rate. Inter media synchronisation is concerned with the coordinated

play-back of heterogeneous media. The "lip" synchronisation problem illustrates the need for synchronisation between two time-dependant media, as the timing between video frames of a person speaking and the corresponding audio needs to be preserved at all instants. Finally, the example previously mentioned of a CD-ROM slide presentation with corresponding audio illustrates a mixture of explicit time constraints between time-independent media and time dependant media, in which specific segments of audio recording, possibly of variable durations, need to be played along the corresponding slide.

3.2.2 Temporal Intervals-based Models

In temporal intervals based models, a time interval constitute the fundamental logical data unit (LDU). A time interval $\mathcal{I}$ is defined by its endpoints: starting time t_s and ending time t_e, with t_s strictly less than t_e, $(t_s < t_e)$. Time can often be represented through the intuitive and familiar notion of the timeline: a temporal axis punctuated by certain events occurring at specific moments and with precise durations. However, in many real life applications, this simple abstraction will suffer from serious drawbacks, as assigning absolute time values to all events may not be possible. In an interactive application, a user may arbitrarily decide to pause a certain stream, then manipulate an item for a certain duration before restarting the paused stream, and this information can not possibly be known in advance. However, some sort of implicit ordering between two media items may still be desired in *relation* to each other (e.g. show slide s_1 before playing audio au_1...) without the need for absolute time references, but rather according to some condition being fulfilled (...whenever the user clicks a certain button). The second drawback of absolute time specification is that if a certain value needs to be modified, all values in the system must be updated accordingly, making it very expensive to maintain. These shortcomings naturally led to the development and formulation of more flexible temporal models in which it is possible to express *partial* information about events without the need for absolute time specification.

Allen's Temporal Intervals

Relative temporal information between time intervals were first formalised by the work of Allen (1983) in which the possible combinations between the endpoints of two intervals, using the set of operators $\{<, =, >\}$ (*before, equal, after*) determine a number of mutually exclusive relations between these intervals. These relations are *before, equal, meets, overlap and during*, with *starts* and *finishes* as additional special cases of the *during* relation. These interval temporal relations are illustrated in Figure 3.1 along with the corresponding endpoints specifications, where s_1, e_1, s_2 and e_2 are respectively the start and end time of two temporal intervals. With the exception of the *equal* relationship, these temporal relations are invertible ($\mathcal{I}_1$ can be *before* or *after* $\mathcal{I}_2$, etc) leading to a total of thirteen possible relations between any two intervals.

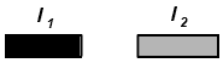
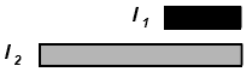
Relation	Symbol	Example	Endpoints Specification
I_1 *before* I_2	$<$		$e_1 < s_2$
I_1 *equal* I_2	$=$		$(s_1 = s_2)$ & $(e_1 = e_2)$
I_1 *meets* I_2	m		$e_1 = s_2$
I_1 *overlaps* I_2	o		$(s_1 < s_2)$ & $(e_1 < e_2)$ & $(s_2 < e_1)$
I_1 *during* I_2	d		$(s_2 < s_1)$ & $(e_1 < e_2)$
I_1 *starts* I_2	s		$(s_1 = s_2)$ & $(e_1 < e_2)$
I_1 *finishes* I_2	f		$(s_2 < s_1)$ & $(e_1 = e_2)$

Figure 3.1: Temporal Relations between two media intervals (according to Allen (1983))

We will be using Allen relative temporal interval representations for information modelling purposes extensively throughout the rest of this thesis.

Enhanced Interval-based Model

One of the limitations of Allen's Interval relations is that they implicitly require the knowledge of interval endpoints positions in *relation* to one another. As an example, it could be the case that the media items $\mathcal{I}_1$ and $\mathcal{I}_2$ need to start at the same time, while how these intervals end in relation to each other is of no importance. In interactive multimedia applications, it will often be infeasible to specify all relative positions of media interval endpoints. As a result, this is a serious limitation to the expressive power of Allen's relative intervals. In order to address this issue, Wahl and Rothermel (1994) introduced an extra operator to the three operators $\{<, =, >\}$ used to compare interval endpoints. The "?" operator simply represents *any* of the previous three (endpoint can be either before, equal or after another endpoint), in other words in describes unknown relative position between two endpoints. By applying all the possible combinations of these operators to intervals' endpoints, Wahl and Rothermel identified ten corresponding interval operators. These are illustrated in Figure 3.2, in which the symbol δ_i represents the duration between two specified endpoints of the respective intervals and a hatched interval ending representing unspecified relations between endpoints.

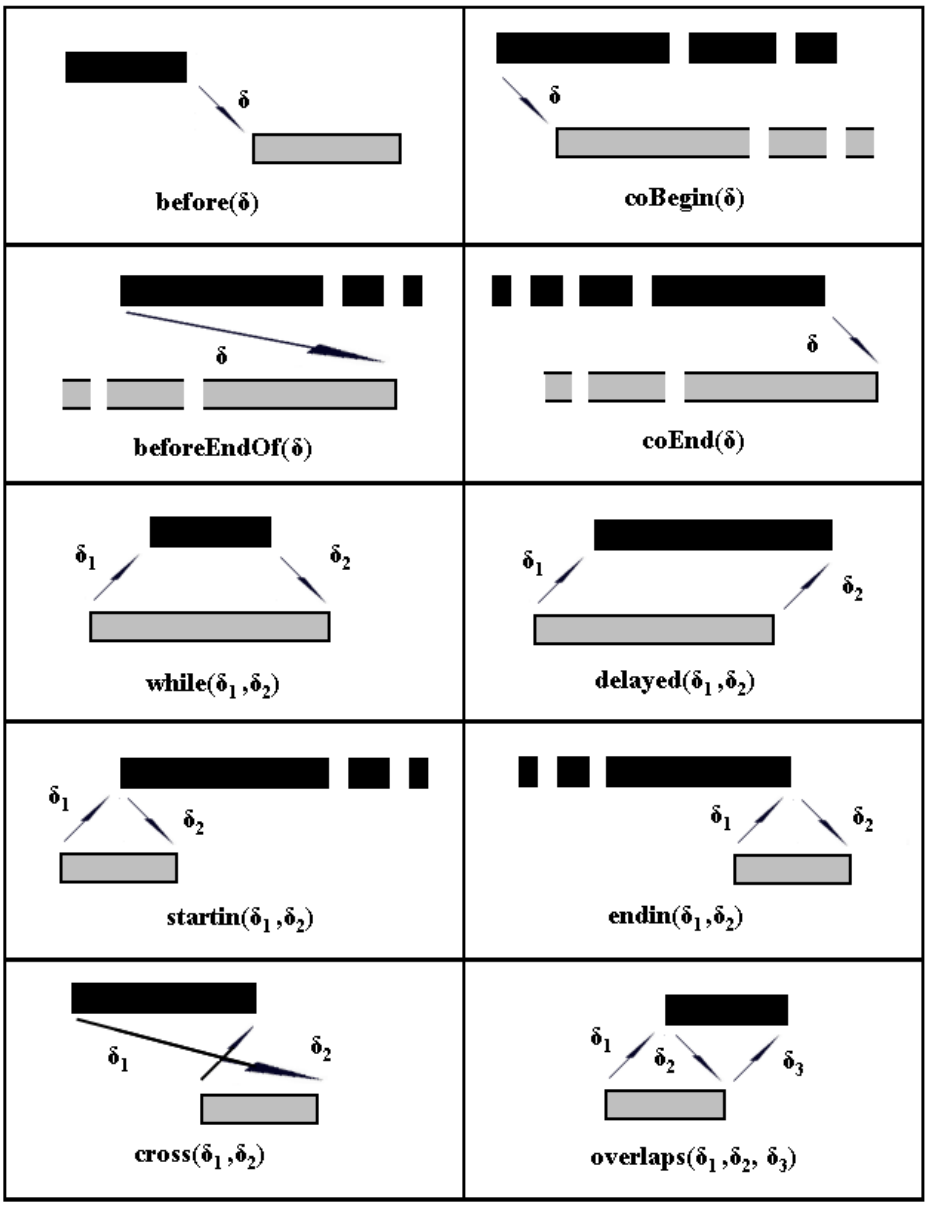


Figure 3.2: Enhanced Intervals Relations between two media items (adapted from Wahl and Rothermel (1994))

Timed Petri Nets

Petri nets are used to model systems which exhibit a combination of asynchronous, parallel and concurrent activities (Peterson 1977) and can be represented as directed graphs with two types of nodes: a set of conditions called *places*, a set of possible activities called *transitions* and directed arcs which run between places and transitions. An arc directed towards a node is an *input* (pre-condition) and a arc leaving a node is an *ouput* (post-condition). When all the conditions of a specific transition are met, the transition is said to be *enabled*. By *firing* the transition, the system moves from the input places state to the output places state. Figure 3.3 illustrates a *marked petri net*, where a *token* represents a condition being met.

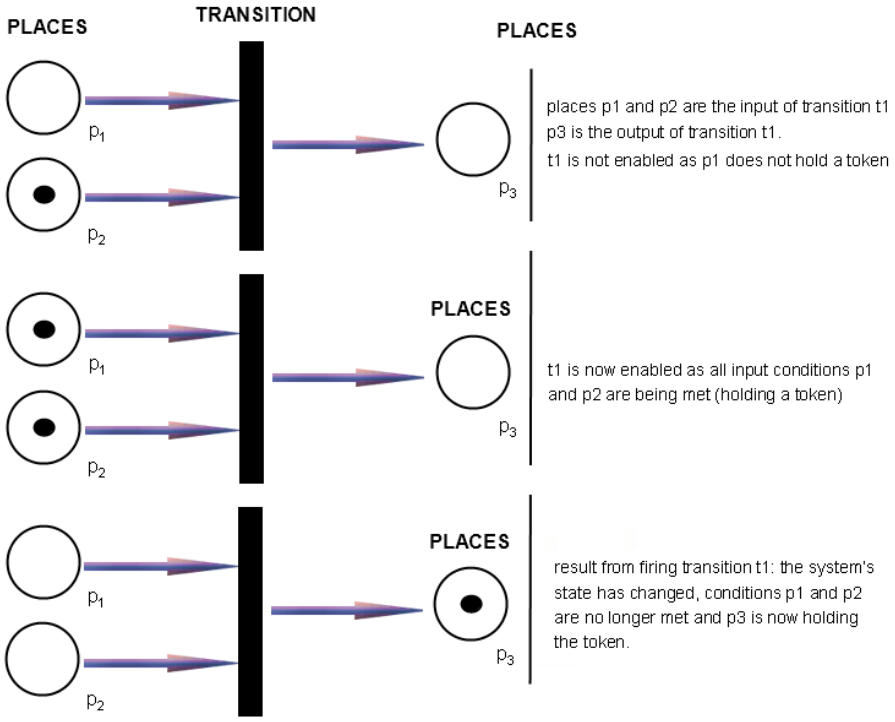


Figure 3.3: Simple Marked Petri Net

Event durations are not explicitly specified in the previous model: the firing of transition is considered to be instantaneous and the duration in which the system remains in a given place is undetermined. As a result, an extension of the previous model, called Timed Petri Net (TPN), can be used to model multimedia systems (Little and Ghafoor 1990). In the timed model, the system remains in a given place for the specified duration τ, a transition's execution remains instantaneous but it's duration can be modelled by introducing an delay place P_δ of duration δ. Figure 3.4 illustrates the Time Petri Nets representations of corresponding Allen's temporal intervals.

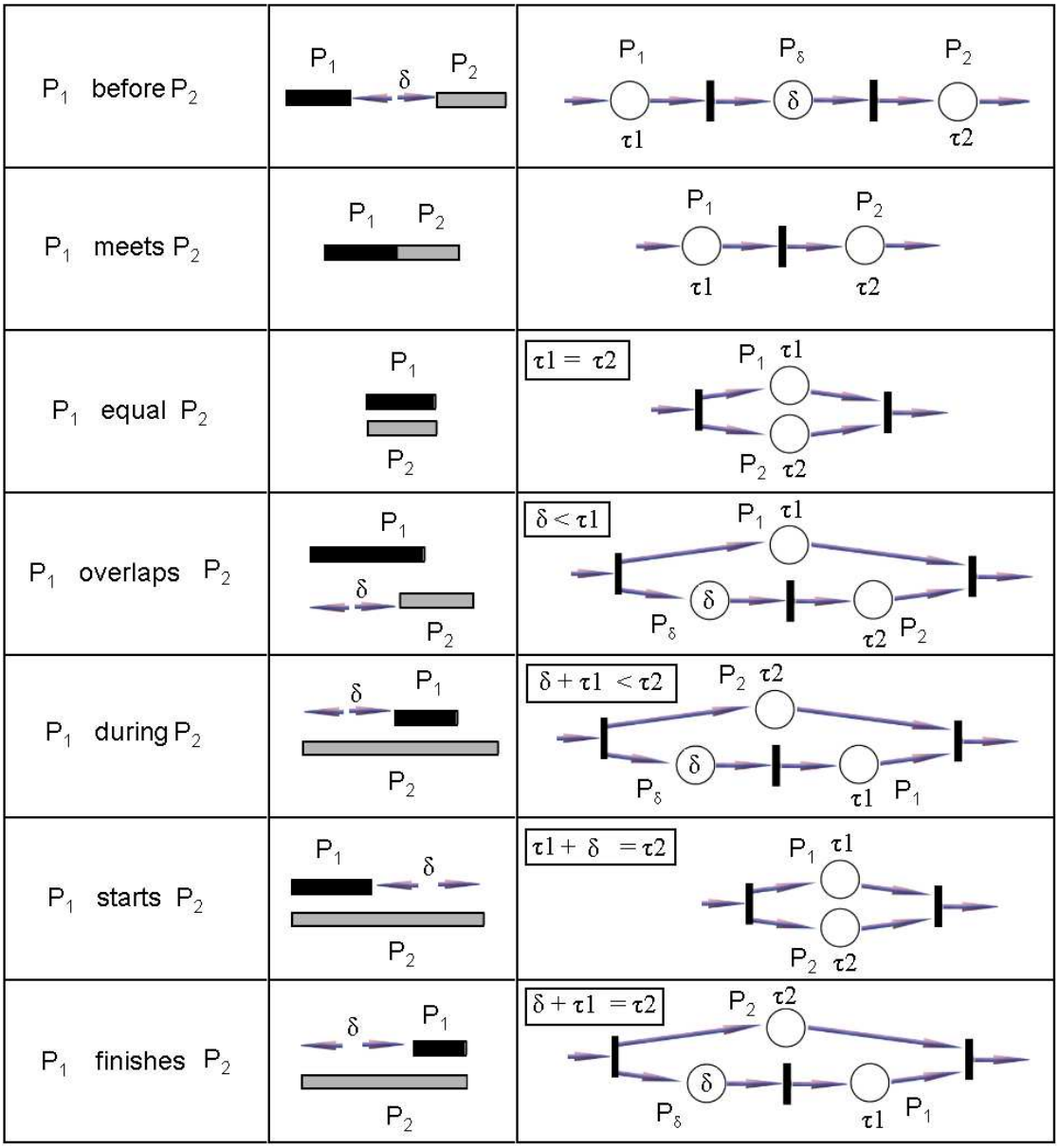


Figure 3.4: Timed Petri Nets and corresponding Allen Interval representations (according to Little and Ghafoor (1990))

Fuzzy Temporal Intervals

The temporal interval representations presented so far have all implicitly assumed a Boolean temporal membership to individual intervals: either an event belongs to a specific interval (within its' time boundaries) or either it is not. This is the situation depicted by Figure 3.5 a.): if an event occurring at time t_e is such that $t_e \in \{t_1, t_2\}$, then it belongs to interval I_1, otherwise, it does not. However, there are many cases where such a constrained interpretation of temporal membership does not well represent the needs of real life scenarios. In these cases, fuzzy temporal intervals can be used to model changes which are *gradual* rather than abrupt.

Let M(t) be the membership function, which value between 0 to 1 describes the level of membership to an interval, according to time t. In the standard interval representation, M(t) can only be one of 2 discrete values: $\forall\ t,\ M(t) \in \{0, 1\}$. In a fuzzy temporal interval representation, an interval membership function will typically be represented by M(t) = f(t) if t $\in \{t_1, t_2\}$, M(t) = 1 if t $\in \{t_2, t_3\}$, M(t) = g(t) if t $\in \{t_3, t_4\}$ and finally, M(t) = 0 if $\notin \{t_1, t_4\}$, where f(t) and g(t) are generally continuous function, with f(t) gradually increasing from 0 to 1 and g(t) gradually decreasing from 1 to 0. In the Figure 3.5 b.), the membership function for interval I_1 thus takes the following values according to t: if t $\in \{t_{11}, t_1\}$, $M(t) = f(t) = \frac{1}{t_1 - t_{11}} \cdot t - \frac{t_{11}}{t_1 - t_{11}}$ (equation of line going through the points (t_{11},0) and (t_1,1)), if t $\in \{t_1, t_2\}$ then M(t) = 1, if t $\in \{t_2, t_{12}\}$, $M(t) = g(t) = -\frac{1}{t_{12} - t_2} \cdot t + \frac{t_{12}}{t_{12} - t_2}$ (equation of line going through the points (t_2,1) and (t_{12},0)), and finally M(t) = 0 if t $\notin \{t_{11}, t_{12}\}$. One of the most interesting aspects of using fuzzy temporal intervals is that an event t_e can partially be a member of 2 distinct intervals at the same time. In the example of Figure 3.5 b.), this is the case if $t_e \in \{t_{21}, t_{12}\}$.

We will be using fuzzy temporal interval representation for information modelling purposes in our own meeting browsing implementation, later in this thesis in Chapter 7, section 7.2.2.

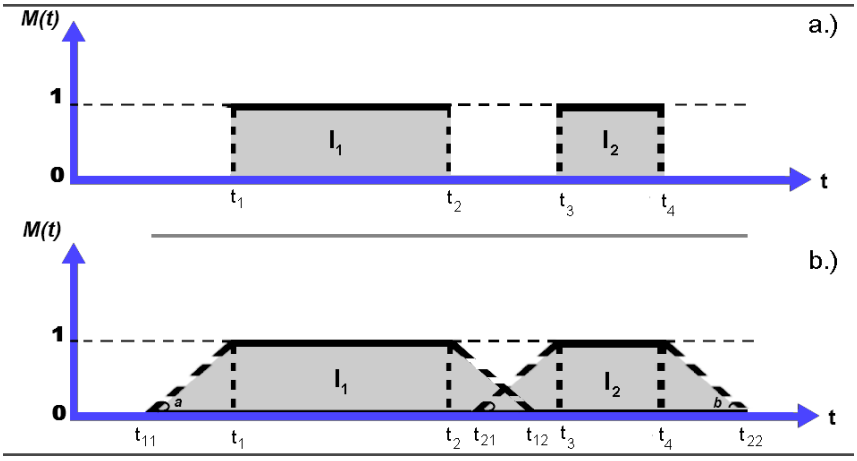


Figure 3.5: Standard (a.) v.s Fuzzy (b.) Temporal Interval Representations

3.2.3 Event-based Models

Firefly Document Model

Firefly (Buchanan and Zellweger 1992) is a multimedia document authoring system designed to implement optimal synchronisation between various media items within a multimedia document, given a user-specified set of constraints. The system automatically generates media items presentation schedules, which involves in certain circumstances stretching or shrinking media segments. The Firefly document model consists of three items: the *media* item, which describes the type and content of medium, the *temporal synchronisation constraints* and an *operation* list, which defines operations associated with particular events. The temporal synchronisation constraints are defined according to specific points of interest occurring during the multimedia presentation, called *events*, and which can be selected through the interface by previewing a media item. The constraints ruling the ordering of events run from a *source event* $\mathcal{E}_{src}$ to a *destination event* $\mathcal{E}_{dest}$ and are specified using *temporal equalities* and *temporal inequalities*. Temporal equalities are used when two events either occur simultaneously or when $\mathcal{E}_{src}$ precedes $\mathcal{E}_{dest}$ by a predetermined amount of time. Temporal inequalities are used when temporal constraints between two events are only partially specified (i.e: $\mathcal{E}_{src}$ precedes $\mathcal{E}_{dest}$ by at least or no more than time τ). The Firefly system provides a graphic view of the synchronised media, as illustrated in Figure 3.6 in which squares represent start and end of a media item and a circular node represents an internal event of interest, and the floating graph at the top right hand of the figure represents an asynchronous user triggered event.

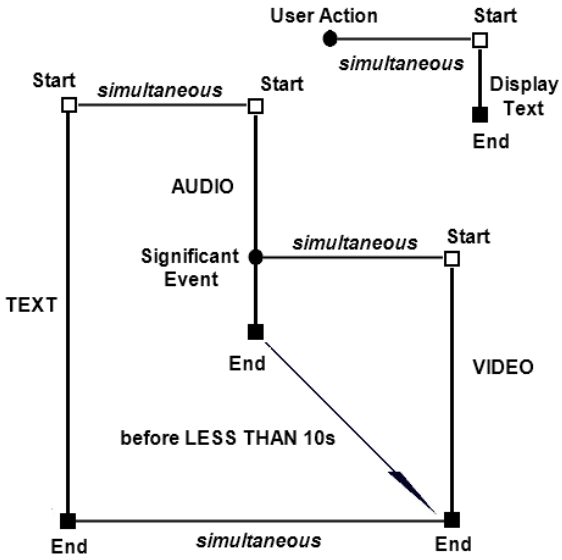


Figure 3.6: Event based temporal view of multimedia document (adapted from Buchanan and Zellweger (1992))

3.3 Multimedia Specification Models

In order to produce multimedia documents and applications portable across a vast number of heterogeneous platforms, without the prohibitive need to generate various incompatible versions, standard specification models have to be developed. More importantly these models need to be adopted by applications designers and multimedia document authors. Specification models aim at providing a structure and detailed temporal behaviour information of interactive multimedia document, while enforcement of the timing constraints between the various elements lies with the application-specific synchronisation mechanims.

3.3.1 HyTime

The HyTime (Hypermedia/Time-based Structuring Language) standard is an application of the SGML (Standard Generalized Markup Language) and provides mechanisms to specify links between various information objects (hyperlinks) as well as spatial and timing information of multimedia presentation and hypermedia data (HyTime 1997). Temporal information can be absolute or expressed in relation to other events, thus all the temporal interval relations described in 3.2.2 can be specified using Hytime. Complex specification, the effort needed for specification maintenance, and a lack of tools, systems and examples relying on HyTime may be some of the reasons why this standard has not been widely adopted (Blakowski and Steinmetz 1996, Rutledge et al. 1999).

3.3.2 MHEG

MHEG (Multimedia and Hypermedia information coding Expert Group) (MHEG 1991) is an ISO (International Organization for Standardization) standard produced by the MHEG working group, which purpose is to define the structure, item relationships and timing dependencies between various multimedia items in a multimedia document. MHEG is an object-oriented model and thus, represents every single media items within a multimedia document as instances of MHEG *objects*, with type description, spatial and temporal information, and either data content or a reference to the actual data (Meyer-Boudnik and Effelsberg 1995). MHEG specifies the possible behaviour of objects according to an allowable set of *actions* depending on the object type. Complex objects can be created by combining simple objects. Interaction is supported at runtime as one source object can register links to other target objects. A change of state of the source object will then trigger a corresponding set of actions in the target objects. More complex specifications can also be expressed as conjunctions of state changes in several source objects. Play-back at a presentation terminal is performed by an MHEG *engine*, whose responsibility is to correctly interpret the multimedia document MHEG object specification, ensure quality-of-service, etc. As is the case of HyTime, complex specifications and a lack of tools and publicly available source code for MHEG engines

may be have been responsible for the low impact of the standard. The MHEG working group has now ceased its activities.

3.3.3 SMIL

SMIL (Synchronized Multimedia Integration Language) (SMIL 2001) is an extension of the XML (Extensible Markup Language) language which permits specification of the dynamic behaviour of individual elements within an interactive multimedia document. In addition to describing layouts and hyperlinks, SMIL can specify the temporal behaviour of the various elements using implicit timing according to the document elements' composition (Rutledge 2001). Elements can be played sequentially ($< seq >$) or in parallel ($< par >$) while temporal behaviour are defined by the relationships between composite elements (as previously described in the interval-based models in 3.2.2). Delays can be specified with the $< begin >$ and $< end >$ attributes, while an absolute duration value can be set with the duration $< dur >$ attribute. An element duration set to "$indefinite$" requires user intervention before triggering the subsequent temporal behaviour of the next element. Hundreds of different types of fading transitions can be explicitly specified.

3.4 Multimedia Content Models

The temporal and synchronisation models surveyed so far in this chapter are merely concerned with timing and synchronisation issues, while content description of data within multimedia document is essentially ignored. With the explosion of digital information now available on the web and public and private institutions' databases, there has never been such an urgent need to add structure and content description to stored digital data. In what follows, we briefly examine the MPEG-7 standard and the semantic web.

3.4.1 MPEG

The MPEG Compression standards

Quite interestingly, given the focus of this thesis, efforts to develop digital video compression standards seem to have originated with research in teleconferencing and video-telephony applications, motivated by the need for real-time video transmission with minimum delay (Liou 1991). More generally, digital video compression standards are necessary to ensure interoperability of codecs (coder-decoder) used across a large variety of digital video equipment.

The MPEG (Moving Picture Experts Group) (MPEG 1988) family standards are compression and interchange formats for motion pictures and audio, and are the result of efforts in digital video compression standardisation started in 1988 by the ISO MPEG working group (Gall 1991). The MPEG standardisation efforts have been hugely successful and is now the *de facto* standard

in digital video compression. The MPEG standard specifies how video and corresponding audio streams can be compressed and synchronised into a single bit stream. Video compression in MPEG is achieved by using a combination of motion prediction in the temporal domain (across successive images) and redundancy in the spatial domain (across a single image). In addition, there are three levels of compression for the corresponding audio, with the higher levels providing higher audio quality but requiring more processing resources. Audio compression without perceptible loss is achieved through a combination of perceptual coding (the frequencies of the acoustic signal which can not be perceived by a human ear are discarded), redundancy (in the acoustic signal) coding, frequency masking (sub-bands of the frequency spectrum where the signal is the strongest hide nearby sub-bands with weaker signal) and temporal masking (exploits the ear's delay in adjusting to sudden shifts in sound level).

MPEG-4

One of MPEG-4 (Koenen 1999, MPEG-4 2002) stated goals was to provide a uniform multimedia encoding and decoding standard before the emergence of a multitude of incompatible proprietary formats and media players. MPEG-4 uses an audio-visual *scene* paradigm to represent media content as a hierarchical composition of primitive *media objects* such as images, audio and video objects. MPEG-4 was designed to support highly interactive applications, thus media object description allows (at the scene's author discretion) individual object temporal and spatial scaling and manipulation, independently of background. The Binary Format for Scenes (BIFS) describe spatial and temporal relations in a scene and allows the modification of properties and attributes of an object without affecting the object itself. As a result, a large number of interactive operations are possible such as random navigation through a scene as well as generating custom viewing, moving objects into different positions, etc.

MPEG-7

MPEG-7, formally known as the Multimedia Content Description Interface, is a standard for creating generic multimedia content *description* (Martinez 2002, Martinez et al. 2002, MPEG-7 2004), independently of the underlying media coding, storage or transmission technologies. The main elements of the MPEG-7 standard consists in *Descriptors*, *Description Schemes* (DSs) and a *Description Definition Language* (DDL). Descriptors are used to describe low level features such as frequency, amplitude, energy, phase (audio features) or color, texture, shape and motion (visual features). Description Schemes (Salembier and Smith 2001) describe higher-level audio-visual elements by specifying relationships between the various description components (Descriptors and other Description Schemes). DSs can be used to describe content management information (media creation, format, quality, classification), temporal information, content description on a structural level (spatial and/or temporal properties of audio-visual *segments*) or semantic level (events, ob-

jects, concepts, places). DSs can also describe summaries, highlights and other various decomposition of the audio visual content. With the Description Definition Language (Hunter 2001) users can define their own Description Schemes and Descriptors. The DDL is based on the XML (eXtensible Markup Language) schema language (XMLSchema 2000) with MPEG-7-specific extensions.

3.5 Concluding Remark

While this chapter has only partially covered the vast area of multimedia modelling, as an exhaustive review would be both infeasible and beyond the scope of this thesis, it shall nevertheless provide the reader with a succinct overview of a number of important existing multimedia modelling concepts, many of which will be used throughout the rest of this thesis.

Temporal intervals are the most common representation in multimedia synchronisation models. Allen's original work on relative intervals was subsequently extended to increase their expressive powers (undefined endpoint positions in enhanced interval model, partial membership in fuzzy temporal intervals, etc.) Once an active research problem during the 80's to mid 90's, multimedia synchronisation has evolved from a modelling perspective to practical system implementations. In the process, several specification standards were proposed, many of which (HyTime, MHEG) failed to have any lasting impact due to over complex specifications and lack of implementations and supporting tools. The MPEG working group has successfully proposed a number of compression and synchronisation standards which are now widely used. As multimedia synchronisation issues have been resolved, the research focus has now moved on to developing multimedia content description models such as the one proposed by MPEG-7, in order to improve manipulation and retrieval of multimedia information.

Chapter 4

The Meeting Artefact History Model

We saw in Chapter 2 that although meetings produce information not readily available in other multimedia recordings, namely the interactions between participants, this additional source of information has so far seldom been specifically incorporated as a browsing modality in existing meeting browsing tools. A notable exception is to be found in the work of McCowan et al. (2005), who propose using low-level audio and visual features (speech activity, energy, pitch and speech rate, face and hand blob) to model meetings as a continuous sequence of high-level meeting group actions (monologue, presentation, discussion, etc) using a HMM model based on the interactions of individual participants. In Chapter 3, we saw that considerable efforts have been made in creating formal models for the representation of temporal and content models of multimedia data. Even though, content description standards like MPEG-7 permit the description of interaction information, to the best of our knowledge, no formal model has been proposed which explicitly tackles the issues of *capturing and managing* meeting interaction information *in real-time*, for the specific goal of facilitating later post-meeting information access to meeting recordings.

In this chapter, we present a novel multimedia model which is specifically targeted at computer-mediated meetings which main purpose is the collaborative production of meeting *artefacts*. Even though this may sound somehow limiting, many real-life scenarios involve meetings which are indeed targeted at the production of some sort of meeting artefacts: work-plans, presentations, summaries, and the endless number of projects and activities which fall under the remit of *collaborative design* (Anupam and Bajaj 1994, Bafoutsou and Mentzas 2002, van Leeuwen 2003). In these scenarios, certain self-contained information items (e.g. an object) are likely to be regularly manipulated. The semantics associated with these information items and their individual manipulations are dictated by (i) the scope (or limitations) of the applications in use during the computer supported meetings and (ii) the particulars of the design process. In an online, artefact-focused meeting scenario,

computer mediation permits the generation of low-level interaction metadata in real-time, without the need for any recognition technologies. By capturing and timestamping participants' non-verbal interactions with certain meeting items, one can associate these semantic objects with persisting histories. This information, which is usually lost in common multimodal meeting settings, offers new possibilities for accessing meeting recordings. Potential semantic links between objects, while not necessary obvious when looking at a meeting outcome, can now be uncovered by investigating temporal relationships between meeting objects. As non-verbal actions are sparse, have specific semantics, and are generally associated with significant meeting events, they can also be appropriate for the visual scanning of recordings.

In what follows, we present a generic interaction-based artefact history model and novel time-based paradigms for retrieving information from *artefacts focused meetings* (Bouamrane and Luz 2006b), which will constitute the main theoretical foundation of this thesis.

4.1 Artefact-Focused Meetings

In a specific scenario, which we will refer to as *artefact focused meetings*, one or many documents such as texts, sketches, drawings, plans are either mentioned or produced during the meeting, either to support the decision-making process or in some cases, as the *goal* or focal point of the meeting, as for instance in collaborative design.

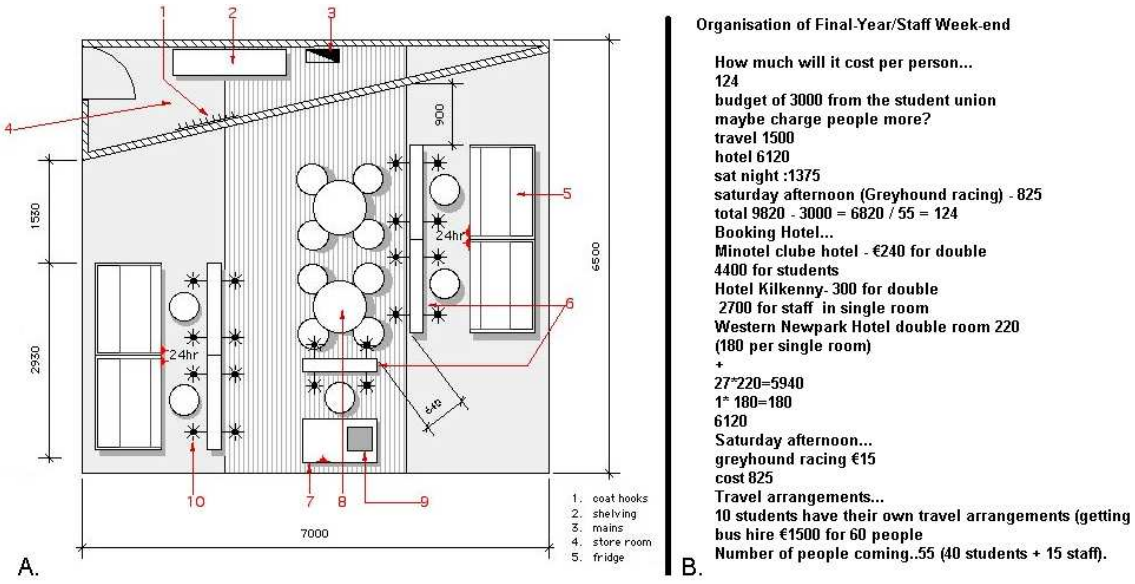


Figure 4.1: Two examples of artefacts produced as outcomes of collaborative activities

Figure 4.1 illustrates two examples of possible outcomes of artefact focused meetings: a plan and a text document. Although the outcome is the obvious *product* of the collaborative meeting activities, it offers no clue about the often laborious *process* by which it was achieved. To illustrate

this, consider the text of Figure 4.1b.: it is a one sided A4 page document, yet it is the result of a collaborative writing task involving more than two hundred edits performed on a basic text skeleton. The document production process is either lost (outside the meeting participants' individual memories) or needs to be recorded in a content rich continuous medium such as audio or video. In the latter case, the problem now consists in accessing relevant parts of the content rich media. As it is a common workplace practice to have many ongoing projects at the same time, some of which might be put on hold while some condition outside the remit of the office is resolved, specific information could be suddenly required months after the last meeting took place. Participants of the meeting will have by then likely forgotten the details of the meeting, or the information could even be sought by someone who did not attend the meeting at all. While the meeting outcome will certainly contain relevant information, one will still be left wondering what were the reasons which motivated certain choices:

- *Does this choice of material comply with fire regulations?*
- *Why did they decide to book a hotel outside of town?*

To answer these questions without listening to the entire recording, one needs efficient means (indexing and navigation tools) to access parts of the meeting recording where relevant information is most likely to be found.

4.2 Generic Low-Level Interaction Metadata in Artefact-Focused Remote Meetings

As previously outlined, computer-mediated, artefact-focused meetings can permit the generation of low-level interaction metadata in real-time, without the need for any higher-level (speech and gesture) recognition technologies. We propose the Archer & Target metaphor of Figure 4.2 to illustrate the type of generic interaction metadata which can be generated in remote computer-mediated, artefact-focused meetings. There are as many archers as remote meeting participants, while targets represent either meeting artefacts, or self-contained information items within a larger meeting artefact. The archers (participants) may use several types of bows (actions) and use different types of arrows (content of actions) at different times (clock), depending of their current focus (targets). Our intuition is that by collecting this rich set of interaction information, post-meeting access to meeting information will be possible on a variety of levels, such as looking at the content of actions (time and type of arrows) or by analysing the targets (who targeted it, with what and when, for how long?), etc.

The following list 4.1 describes the nature of generic low-level interaction metadata which can be generated in remote computer-mediated meetings.

List 4.1 Generic low-level interaction metadata in computer-mediated meetings.

→ *Agent (identity of participant who perform the action)*

→ *Type of Action*

→ *Content of Action*

→ *Target of Action*

→*Time & Duration of Action*

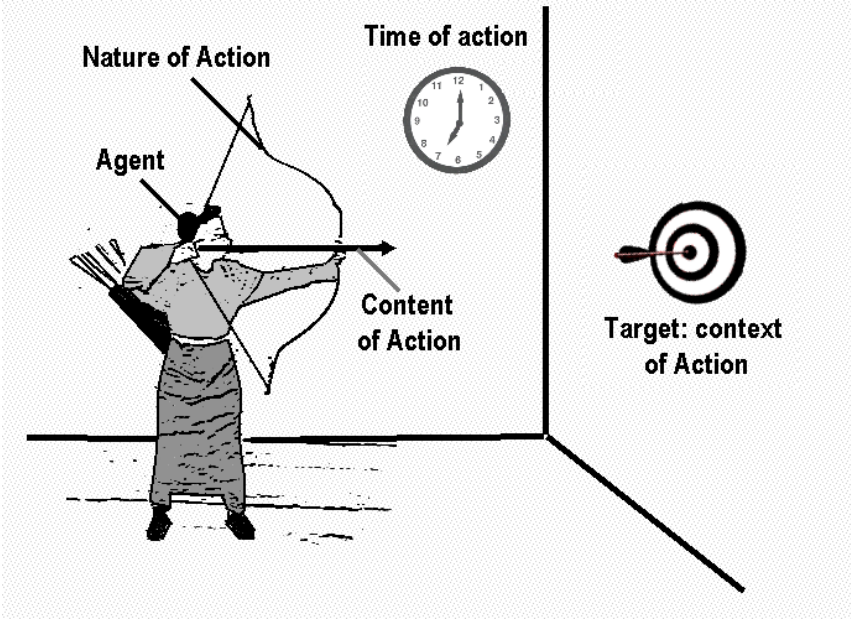


Figure 4.2: The Archer & Target metaphor for interactions in remote meetings

4.3 Artefact History Model

In the previous Archer & Target metaphor, while it is fairly clear to understand what the archer stands for (participant), defining exactly what the target represents can be more problematic. We have mentioned meeting artefacts, or more precisely, self-contained information items of a meeting artefact. In this section, we define important concepts of our artefact-history model.

Gibbs et al. (1994) define an artefact as the objects produced by a particular medium such as prints, audio and video recordings, etc. and refer to *media object* as the machine readable representation of these artefacts. The media object can be a representation of a *natural* artefact (e.g. digital image obtained by scanning a picture) or of a *digital* artefact (e.g. computer-generated graphics).

4.3.1 Key Concepts

We envisage a meeting scenario where geographically dispersed users interact with a "document" (text documents, drawings, tables, plans, etc.) which acts as the focal point of the meeting. Participants can communicate through continuous media communication channels (audio and/or video). Furthermore, it is assumed that interactions with the spaced-based document are computer mediated and can be automatically detected and recorded. We define the following key terms:

Document The set of artefacts used during the meeting. This is to a large extent the focal point of the meeting, either because the document supports the decision making process or is the meeting intended final outcome (work plans, technical drawings). The document can be collaboratively written text, slides for a presentation, graphs, drafts and plans, for collaborative design, audio or video clips, or any combination of these.

Data Objects Within the document, smaller data objects which can be treated and manipulated as individual semantic entities. The granularity of the semantic data objects is best defined according to the application scope. To illustrate this point, in common applications, a pixel or character would have no intrinsic semantics as opposed to a word, sentence, paragraph, a shape or image.

Primitive Operations Primitive operations will typically modify some property of a data object. Examples of primitive operations are *insertion*, *deletion* of characters in text, modifying the texture, size, of an image or shape.

Manipulation Rules Manipulation rules need to associate an unambiguous and definite outcome when a data object manipulation affects other objects. Examples are *cutting* and *pasting* paragraphs or shapes, occlusions, etc.

Timestamp A timestamp records information about all primitive and manipulation operations previously defined. Information recorded are: the *agent* who performed the operation, the *nature* of the operation, the *time* of operation and unless unfeasible, the exact *content* of the operation. Note that this is often partially the case in many existing applications: the nature and content of a number of past operations are stacked for undos and in collaborative applications, some form of timestamping generally needs to be implemented to address concurrency and consistency issues (Ellis et al. 1991, Sun et al. 1998, Sun and Chen 2002).

Object Log For each object, a list of all timestamped actions (primitive and manipulation operations) which affected the object from creation throughout the meeting.

4.3.2 Object-based Retrieval

Common approaches to building visualisation and retrieval interfaces for browsing multimodal meetings emphasise linear access (whether sequential or random) due to the structuring role time naturally plays in multimedia data. Segmentation and indexing according to some features of the time-based media (speaker transition, shot detection) are used to define a number of media intervals. Access to specific media intervals is provided by some persistent representation of the time-based media (keyword, speaker identity, keyframes). By synchronised play-back of multiple media streams, a number of browsing systems (Brotherton et al. 1998, Geyer et al. 2003, Wellner et al. 2004) will ensure that non-verbal artefact manipulations concurrent with the current media interval will be visible to the user. In Figure 4.3[1] selecting the media interval I_2 will not only play the corresponding audio and video but will also display the nature of manipulations on the three objects: O_1, O_2 and O_3 which were modified within the interval duration.

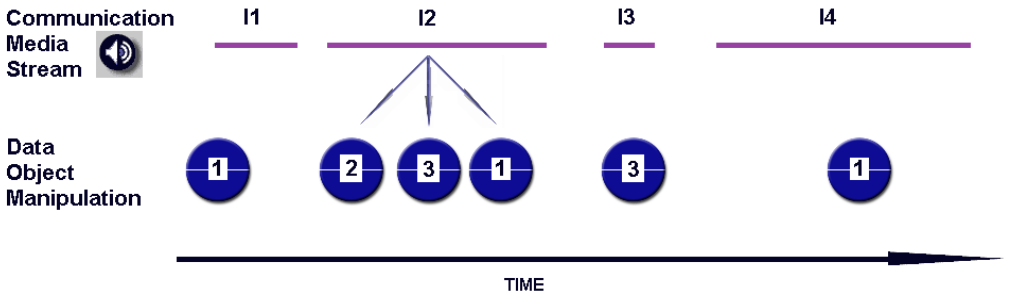


Figure 4.3: The data stream or timeline perspective

However, continuous media such as audio and video, with time as inner structure, are difficult to access for lack of natural reference points, navigation is time consuming and can be confusing, and summarisation is a non-trivial process. In cases of meetings lasting an hour or longer, ASR transcripts, even of good quality, may still represent quite a voluminous amount of information to scan through. Also, as spoken language is significantly different to written language, the transcripts may be difficult to decipher (due to style, repetitions, false starts, etc). Text-based information retrieval techniques can be applied to the ASR transcripts and one could argue that a key-word search can be an appropriate way of overcoming these shortcomings to quickly find relevant information. However, in some cases, someone looking for specific information in a recording (i.e. through a query) and participants who mentioned the relevant information during the meeting may use *different words* to refer to the *same object*. In this case, a key-word search will require additional mechanisms such as a query-expansion using a thesaurus or an ontology, which could

[1] Artefact manipulations have been represented as punctual objects to emphasise their nature as separate abstract entities. In reality, artefact manipulations are time intervals themselves (duration of manipulations)

have unpredictable results in scenarios with significant word error rates.

More importantly, if a meeting's outcome is a document, visual browsing may be a more appropriate way of quickly identifying relevant information. Consider the outcome of Figure 4.1a.: due to the graphical nature of the document, visual scanning is almost instantaneous. In a collocated meeting scenario, one possible way of querying information from other participants would be to simply point at a particular item and say *"what about this thing, there?"* (e.g. *the counter*). Ideally, one would want to have access to *all* segments of the recording where the item in question is *mentioned*. For various reasons, current ASR technology cannot guarantee this will be done reliably. In contrast, we propose a simple alternative: by associating spatial *data objects* (e.g. *the counter*) with a log of all the actions which affected them during the course of the meeting, one can provide access to segments of the recording during which a particular item was *manipulated*. The information retrieval assumption behind such information linking is that object manipulations coincide with a particular object being, at least partially and sometimes briefly, the current focus of the meeting. We refer to this information retrieval paradigm as retrieval from the *data object perspective*.

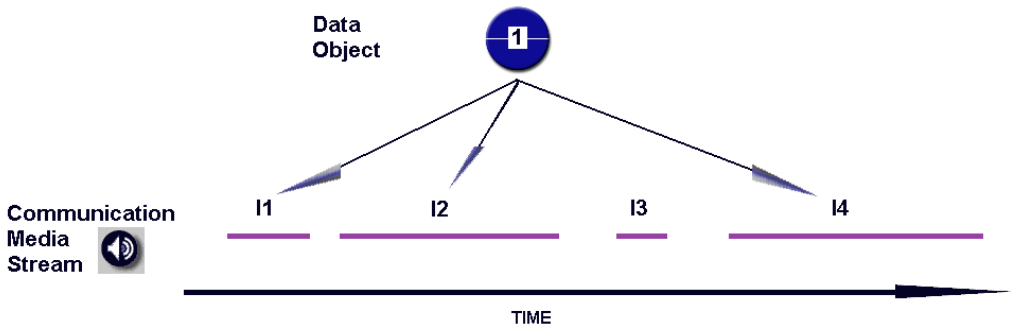


Figure 4.4: Data Object perspective

By logging all (relevant) information relating to the manipulation of a specific artefact, one can link these with all concurrent time-base media segments. Thus, access to the time-based media is now done through selecting a specific *object*, or specific *actions* performed on an object. This is illustrated by Figure 4.4: selecting object O_1 provides access to all time-based segments concurrent with a manipulation of O_1: the three related intervals I_1, I_2 and I_4. This paradigm shift from time to object is rather intuitive, shifting the emphasis from *"what were people doing when they were discussing this (i.e query terms)?"* to *"what were they saying when they did this (manipulate object)?"*

4.3.3 Object Temporal Associations

One immediate property of having access to individual objects' history logs is that it enables us to discover potential associations between specific objects just by investigating the concurrency of actions performed on these objects. The following scenario illustrates how analysing temporal patterns of object manipulation may uncover potential (non-obvious) information. The outcome of Figure 4.1a. is the result of several meetings, each concentrating on resolving a specific issue. The final plan is "*flat*": relationships between the different objects are not a-priori obvious. However, patterns of interactions associated with meeting objects show recurrent manipulations of the *"Table"* objects along with the *"Exit"* item. Listening to audio segments where these objects are manipulated in close time proximity, the user discovers that in this particular project, the client's preferred table layout is not compatible with fire regulations. As a result, the client's *original* layout (no longer visible in the final outcome) had to be modified in order to accommodate the existing *"Exit"* and meet fire regulations, which explain the reasons behind the position of the *"Table"* objects in the *final* layout.

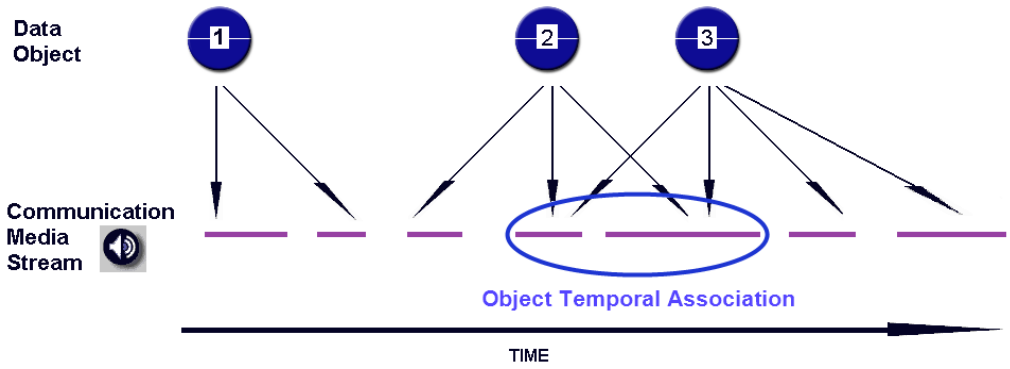


Figure 4.5: Data Objects Temporal Associations

Figure 4.5 illustrates object temporal associations: objects O_2 and O_3 were on several occasions manipulated in close time proximity and as a result share a number of concurrent time-based media segments in their respective history log, indicating a potentially useful information link between these two distinct objects. Object associations enable us to go beyond the information pattern illustrated in Figure 4.4. Specific objects are not only linked to relevant segments of the time-based media, the wider *context* in which a specific object was manipulated during the meeting can be investigated within the context of other data object manipulations.

We thus define the concept of an object's *temporal neighbourhood* (Luz and Masoodian 2005, Luz and Bouamrane 2006) as the set of (i) time-base media segments concurrent with the object manipulation and (ii) actions performed on other objects within the previous time-based segments' duration. We propose the following algorithm for retrieving an object's temporal neighbourhood:

Algorithm 4.1

Object Temporal neighbourhood retrieval.

→ *retrieve the set of all non-verbal actions performed on a specific object*

→ *retrieve the set of all time-based segments concurrent with these actions*

→ *retrieve all actions performed on* different *objects which took place within the duration of the previous set of time-based segments*

→ *iterate through the 2 previous steps until no new actions or time-based intervals can be found*

This object temporal neighbourhood retrieval algorithm is later instantiated in this thesis in our meeting browsing system implementation in Chapter 7, section 7.4.2, where it is applied to paragraphs of a meeting document outcome.

4.3.4 Action-based Browsing

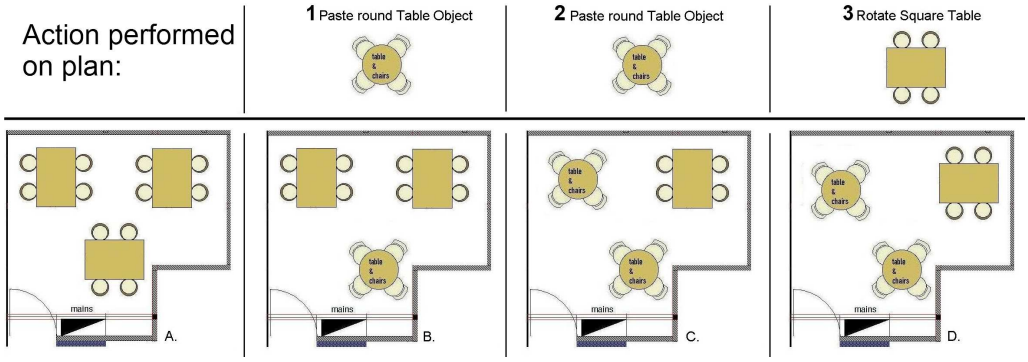


Figure 4.6: Sequence of Actions

We have so far defined spatial objects as potential information retrieval units. Another potential use of a log of non-verbal actions is as a *visual navigation* tool into the time-based media recording. Consider the simple sequence illustrated in Figure 4.6: the upper part of the figure illustrates the evolution of the artefact during the course of the meeting while the bottom part shows the corresponding actions. One might wonder what prompted the choice of two different sets of tables in the final outcome Figure 4.6 d.). By visually observing the sequence of actions, a user may identify the exact moment when an action of *interest* was performed (i.e: pasting round table), thus identifying a region of the time-based media where a potential explanation (*e.g. keep clear escape route*) is likely to be found, as illustrated in Figure 4.7.

For this navigation method to be useful, only certain *significant* actions should be visible during browsing, since for most applications, atomic pixel or character-based operations are meaningless

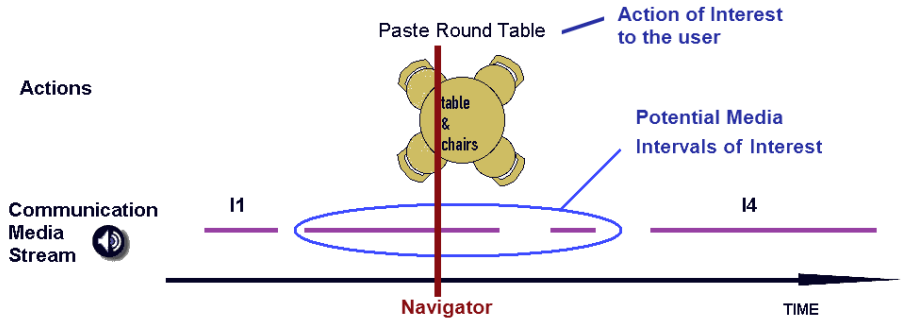


Figure 4.7: Using non-verbal actions as a navigation tool

out of context. Operation filtering can be done at two stages. During the action logging stage, only actions which are potentially useful for information retrieval purposes are captured, while atomic operations are either discarded or buffered (into a broader operation). As previously mentioned, what potentially constitutes a useful action is entirely dependant on the specific application.

Information filtering can also be done at the post-meeting processing stage, where the user can dynamically choose what type of actions he is interested in (i.e: display only *"paste" specific "type" of object*). The definite appeal of such a navigation method is that non-verbal actions will generally have strong associated semantics and are appropriate for quick visual scanning, thus potentially offering a powerful indexing method into the time-based media. Interactions are discrete, generally sparse enough so as not to overload a user with information, and tend to form natural semantic clusters over time (when a specific topic is discussed) allowing for discrimination and segmentation of topics within a meeting recording. This indexing method is also perfectly accurate in timing and content as we envisage low-level automatic metadata generation to be performed without the use for (speech or gesture) recognition technology.

4.3.5 Advanced Query Model

When looking for meeting information, one can use artefact history to perform a search which includes different levels of details. If a user does not exactly know what he is looking for, a general object neighbourhood retrieval may be appropriate. In this type of indexing, all time-based media intervals and object operations related to a specific object are retrieved. Visual action-based navigation, as detailed in the previous section can be appropriate if the information sought is associated with a significant and specific meeting action (e.g. paste round table). Finally, more advanced queries can also be formulated by selecting a type of object combined with conjunction or disjunction of actions and action attributes. An example of such an advanced query is: select all objects *round table*, where operation is *hatch* and attribute is *hatch pattern*. The general form of an advanced query is:

$$\text{retrieve } \textit{object type} \wedge \textit{action type} \wedge \textit{action attribute}.$$

4.4 Summary

We have presented a generic artefact history model for extracting and retrieving information from artefact-focused, computer-mediated meetings. Capturing and timestamping participants' non-verbal interactions with data objects offers new possibilities for meeting information retrieval. Investigating temporal relationships between objects may uncover potential semantic links which are not necessarily obvious when looking at a meeting's outcome. The definite appeal of interaction-based navigation methods is that non-verbal actions will generally have strong associated semantics and are appropriate for quick visual scanning, thus offering a powerful indexing method into time-based media. Interactions are discrete, generally sparse enough so as not to overload a user with information, and tend to form natural semantic clusters over time (when a specific topic is discussed) allowing for discrimination and segmentation of topics within a meeting recording. This indexing method is also perfectly accurate in timing and content as the model we propose produces low-level automatic metadata generation without the use for (speech or gesture) recognition technology. In the following chapter, we describe the implementation of an online speech and text meeting architecture based on the artefact history model described here.

Chapter 5

Implementation: The RECOLED Online Meeting Environment

In this chapter, we describe the RECOLED (RECording COLlaborative EDitor): an integrated architecture for online collaborative meetings which supports speech, text, and "gesturing" activities. In addition, RECOLED records participants' audio exchanges, and automatically generates interaction metadata in real-time. Metadata consists of the logs of users' interactions with a shared text document and also information derived from the use of group awareness widgets (telepointing, free-hand drawing, etc.) Generation and manipulation of histories logs of self-contained elements (paragraphs) of textual artefacts are based on the artefact history model described in Chapter 4. We define rules for the manipulation of paragraph timestamps and illustrate these with several examples. The RECOLED shared-editor design, user interface and usability study described in this chapter were published in Masoodian et al. (2005) while the description of generation and manipulation of interaction information is described in detail in Bouamrane et al. (2005).

5.1 Design of a Remote Meeting Environment

5.1.1 RECOLED: the Collaborative Development of a Collaborative Software

The RECOLED (Bouamrane et al. 2004, Masoodian et al. 2005) remote meeting environment is the result of a fruitful collaboration between Dr. Masood Masoodian and David King of the University of Waikato and Dr. Saturnino Luz and myself, of the University of Trinity College. Design of the final prototype architecture involved all those mentioned above. Dr. Masood Masoodian and David King were involved in the design of the RECOLED online shared-text editor, and are responsible for innovations in the user interface's awareness widgets. Dr. Saturnino Luz was

responsible for developing the RTP (Real-Time Protocol) recording server and participants' speech profile generator. The author was responsible for automatic interaction metadata generation and timestamp manipulation rules which are later described in this chapter.

5.1.2 Choice of Communication Modalities

One of the most common space-based artefacts which is the product of a collaborative activity is a text document. Indeed, studies have shown that a majority of text documents are written involving teams of collaborators (Posner and Baecker 1992). As a result, the RECOLED development team decided to implement a remote meeting environment whose activities would centre on the collaborative production of a text document, similar to existing shared text editors (Ellis et al. 1991, Sun et al. 1997, 1998), albeit with additional innovations in the awareness mechanisms as well as adding interaction logging functionalities. However, as text can be a very slow and constraining communication medium, the shared-text component of the remote meeting architecture needed to be complemented by other modalities, such as "Gesturing" functionalities, in the form of telepointing and free-hand drawing. In addition to the well-documented virtues of providing remote users with awareness functionalities in groupware (Dourish and Bellotti 1992, Gutwin et al. 1996), the choice of adding gesturing capabilities was also designed to augment the scope of possible interactions among meeting participants, and by extension, post-meeting retrieval possibilities. The intuition behind capturing gestures is that in many cases, they are used when participants need to emphasise some *specific* aspect of the meeting document, at some *particular* time, (*e.g. what about this section, then?*, (pointing)), thus potentially indicating important issues covered during the meeting.

Finally, in order to add flexibility and ensure low communication latency, an instantaneous and continuous communication medium, such as speech and/or video was also warranted. In synchronous collaborative work over a shared workspace, audio has been shown to be the most efficient communication medium (Jensen et al. 2000), as meeting information is often essentially audio centric (Li et al. 2000). As we primarily envisaged the meeting architecture as a lightweight environment, which could easily be installed on a personal computer, without excessive need for processing power or bandwidth, it was decided to complement the shared-editing/gesturing environment with speech only.

5.1.3 Choice of Network Architecture

Essentially, two possible types of network architecture were envisaged: a peer-to-peer (multicast) architecture and client-server architecture. The advantages of the peer-to-peer architecture lies in the fact that it can be more resilient in the face of network failure: once the server is isolated or fails in a client-server architecture, the whole systems breaks down, whereas in a peer-to-peer architecture, some participants may still continue to communicate even if others are cut off from

the network. Peer-to-peer introduce however additional implementation difficulties in the meeting architecture, namely in the form of synchronisation of the hosts' clocks and additional complexity for resolving issues relating to concurrency and document consistency maintenance (Ellis and Gibbs 1989, Sun et al. 1998). Two early prototypes along these two possible architectures were developed and tested in parallel in the University of Waikato, New Zealand and Trinity College, Dublin. The complexity of managing meeting interaction history in real-time in these two early prototypes showed that the advantages offered by the client-server architecture clearly outweighed any of the inconveniences cited above, and hence, this architecture was ultimately selected for the final prototype.

5.2 Implementation of RECOLED Meeting Environment

5.2.1 System Architecture

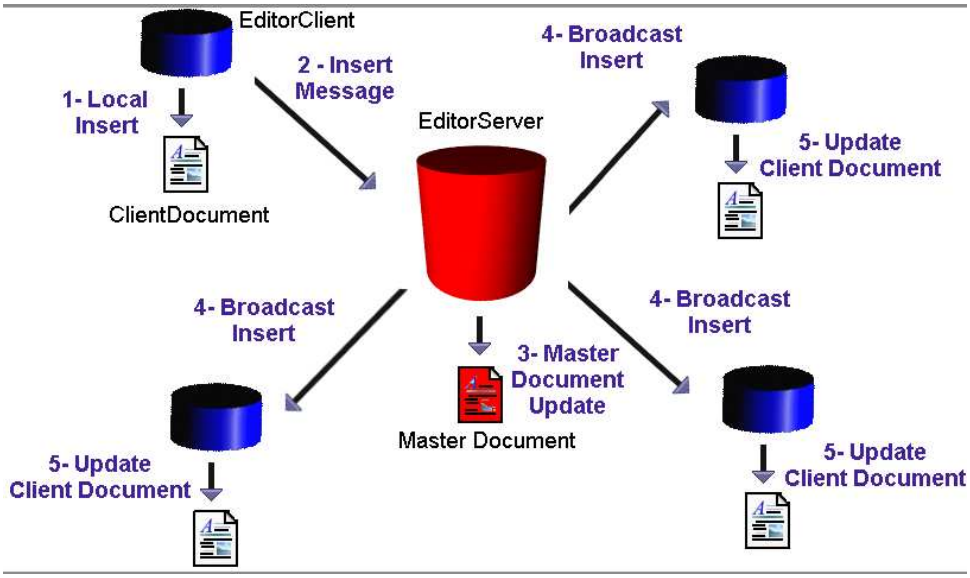


Figure 5.1: RECOLED's Client-Server system architecture

RECOLED is designed in a semi-centralised architecture (Figure 5.1). It consists of a single server and several clients, communicating through TCP-IP (Transmission Control Protocol - Internet Protocol) connections. These components handle the management of collaborative editing and awareness mechanisms. The system is semi-centralised in that clients cannot directly communicate with each other and must relay all communication through the server. The text management server (EditorServer in Figure 5.1) provides concurrency control (Ellis and Gibbs 1989) between clients through an optimistic locking mechanism to ensure consistency across different copies of the shared document. The server also uses its copy of the shared document to provide session persistency by sending new clients an up-to-date copy of the document content when they join the

session. Clients also keep their own copy of the shared document so that the system can have fast response time to local user events. The clients send their local updates to the server as well as receiving other clients' events from the server.

RECOLED was designed explicitly to be used in conjunction with an audio channel. It is assumed that the co-authors will use the audio channel provided by the conferencing tool for their explicit verbal communication, while the implicit communication and awareness will be supported (and recorded) through the shared editor.

5.2.2 Data capture

The information recorded by RECOLED is illustrated in Figure 5.2. On the one hand, all speech exchanges are recorded through the server's RTP recorder. The audio tools used by the clients transmit audio in RTP packets over a datagram socket. The audio is recorded directly from these sockets by an application which keeps the original RTP timestamps and adds arrival timestamps to the recorded data. These timestamps are vital for detecting speech activity and linking it to text segments. In addition, all editing and gesturing operations performed by co-authors are automatically recorded by the server's central timestamping mechanism. This mechanism is subordinated to the locking mechanism and is described in further detail in the following sections.

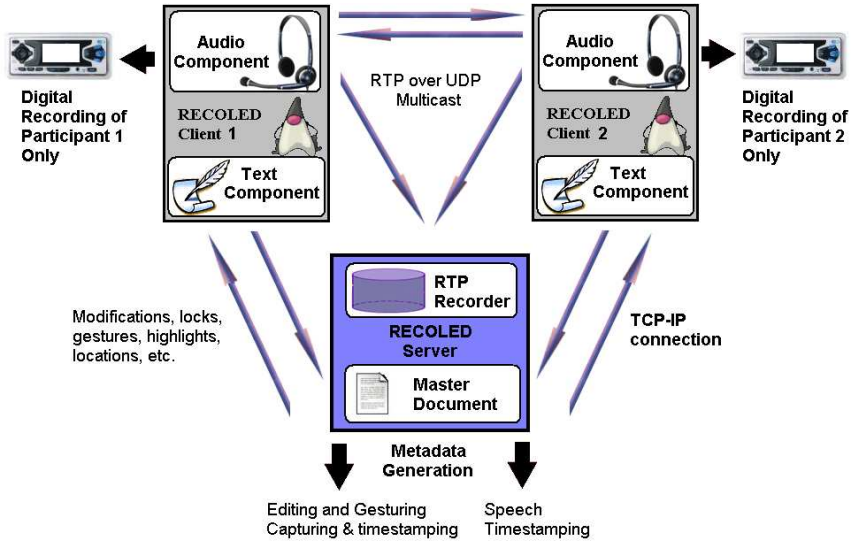


Figure 5.2: RECOLED's audio and editing history recording architecture

5.2.3 Document consistency and Concurrency control

Due to the synchronous and distributed nature of the collaborative writing scenario targeted by RECOLED, concurrency and consistency mechanisms are critical to ensure effective use of the editor. Concurrency control is a well-documented research area in groupware (Ellis and Gibbs 1989, Ellis et al. 1991, Greenberg and Marwood 1994, Sun et al. 1998) and a number of approaches

have been proposed: turn-taking, locking, serialization and operational transformations. After careful consideration, the appropriate granularity for logging information in RECOLED was chosen as a paragraph (although the content of individual actions is also captured in a separate set of action timestamps as described in more detail in section 5.4). Thus, in order to maintain consistency between the copies of the document, we use a document implementation based on paragraphs, so that modifications are made on these rather than on the document as a whole. This segmentation of the document into paragraphs permits concurrent modification of distinct paragraphs, while a locking scheme ensures that a single paragraph can only be modified by one client at a time. This locking scheme is optimistic so that local responsiveness can be maintained. In order to reduce the likelihood of conflicting editing operations, RECOLED uses a number of group awareness mechanisms which are described in details in the next section. In particular, a paragraph currently locked by another participant will have its margin highlighted in that participant's colour (Figure 5.4). A usability study carried out on RECOLED (see Chapter 6) showed that overall, this locking scheme did not hinder the co-authoring task.

Lock manager and timestamping

The lock manager is the cornerstone of our system architecture in that it permits collaboration on a meeting document by managing and preventing participants' conflicting manipulations while also being central to the interaction timestamping process. On the client side, the lock manager is responsible for checking if a participant's edit can proceed. If the participant is already holding the lock, the edit is carried out and sent to the server, which then broadcasts the update to the other clients. However, if the participant does not yet hold the lock for the particular paragraph, then the lock needs to be requested from the server. This process is illustrated in Figure 5.3. On the server side, the lock manager grants or denies a lock depending on the current paragraph's status. Since it is the lock manager's responsibility to permit editing operations, it is also at the centre of the timestamping process. If an update is allowed to go ahead, the lock manager identifies the exact type of operation being performed and triggers the appropriate timestamping operations, generating action and paragraph timestamps, merging or mapping timestamp lists, according to the model described in detail in the following sections.

Character updates and timestamp buffering

Although updates in RECOLED are character-based to allow high responsiveness, generating a separate timestamp for each character typed or deleted would be rather expensive. Furthermore, one of our main goals when designing RECOLED is to identify and capture significant (i.e. meaningful) operations in the document writing process. As such, a character-based timestamping mechanism would be pointless. Therefore, for timestamp generation purposes, a participant's consecutive edits are buffered, and the timestamping is subordinated to the locking mechanism, which

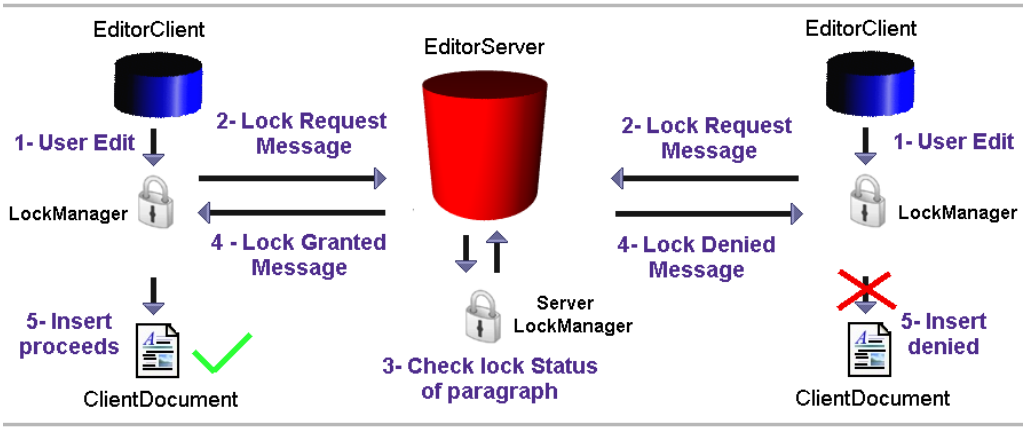


Figure 5.3: The concurrency control mechanism of RECOLED

means that a timestamp is generated only once the paragraph lock is released (currently this is set to 3 seconds of user's inaction).

5.3 RECOLED User Interface & Awareness Mechanisms

Designers of existing groupware applications such as synchronous shared editors have often focused on incorporating interface components which provide means of group awareness, whether it is workspace awareness (e.g. use of tele-pointers, shared scrollbars, etc.) or social awareness (e.g. video links, participant lists or images, etc.). These types of group awareness mechanisms are clearly important in supporting the collaborative work process, and indeed the usability and effectiveness of groupware depends largely on the extent to which it supports group awareness (Dourish and Bellotti 1992, Gutwin and Greenberg 1999). In most synchronous shared editor software, however, the group awareness information is only used to support the collaborative work activity during the synchronous session itself, after which such information is not recorded and largely ignored. Although some systems record information such as the names of the participants, session duration, and other particulars, the information relating to specific user actions such as text insertion, deletion, focus of attention, and pointing, is lost. In order to address this, RECOLED not only provides the users with important awareness information during the collaborative writing and editing sessions, but also extracts and records information derived from the awareness components of the interface.

5.3.1 Group Awareness mechanisms

The editor client provides basic text-editing functionality available in simple single-user editors. In addition, group awareness mechanisms are used extensively in RECOLED to allow users to coordinate their access to the document more effectively. Much of the interface is directed to-

wards providing different types of group awareness. These include avatars, shared scrollbars, text colouring, and support for pointing and gesturing (Figure 5.4).

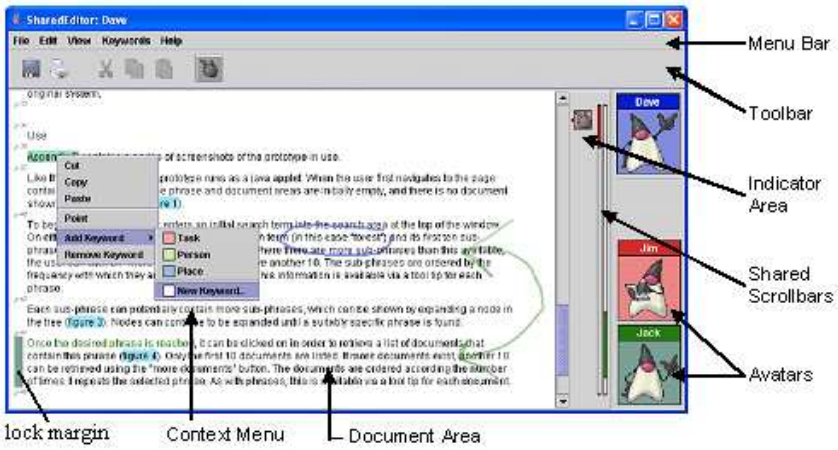


Figure 5.4: RECOLED client application interface

Participants and Actions Awareness

Each RECOLED client displays a set of avatars to the right of the document area. These avatars display the names of all the users currently involved in editing the document, and serve as a key relating each user to a unique colour, which is then used extensively throughout the interface. They also provide a limited amount of action awareness by taking on different appearances depending on the actions of the user (Figure 5.5). These are simply inferred from the types of the latest messages received, if any, and require no extra bandwidth. Author names are associated with the various actions they perform, which are recorded in the document history.

Figure 5.5: RECOLED Client avatars and their actions

Within the document itself, any text that is inserted is initially shown in the colour of the person who typed it, therefore providing ownership awareness. It also provides a limited amount of history awareness by associating the age of a modification with the brightness of text colouring. This ensures that recent modifications stand out from the rest of the document, and thus can be found easily. Coloured text gradually fades into the normal text colour, which ensures that the document will not become distractingly colourful, while allowing users to gain a sense of the order in which characters were typed. Shared scrollbars are shown to the left of the avatars (see Figure 5.4). There is a separate scrollbar associated with each of the other participants, displayed

in their unique colour. These shared scrollbars are similar in appearance and purpose to a user's own scrollbar, but provide location awareness by showing where in the document each participant is located, and view awareness by showing how much of the document they can see. This information can be easily interpreted, and direct comparisons can be made with the user's own scrollbar, which is located next to them in a conventional manner. Users can also choose to synchronize their view of the document with that of another participant by linking their views.

Although this is not done in the current version of the system, information derived from shared scrollbars could easily be incorporated into the document history to indicate on which parts of the document the users focus, as implemented in "read wear" (Hill et al. 1992).

Gesturing

Gesturing mechanisms are incorporated into the editor as a means to support the collaborative writing process and as a complement to audio communication. During a collaborative writing session, authors can perform a considerable number of pointing actions, as highlighted by the figures in Table 6.1 in the next chapter. RECOLED has two different types of gesturing capabilities: simple pointing and freehand drawing. Both actions are recorded as part of the interaction history. To point at something, a user can simply click on that location. This in turn shows the pointed location using an animated radar pulse, with the centre point surrounded by rapidly expanding circles, which captures and focuses other users' attention on the intended point, as shown in Figure 5.6.

Figure 5.6: Telepointing on a paragraph of the Meeting Document

When users need to convey more complex information than a simple location, they can hold the mouse button down and freely draw over the surface of the document. Freehand drawings by a user are then displayed on the screen of every other group member Figure 5.7.

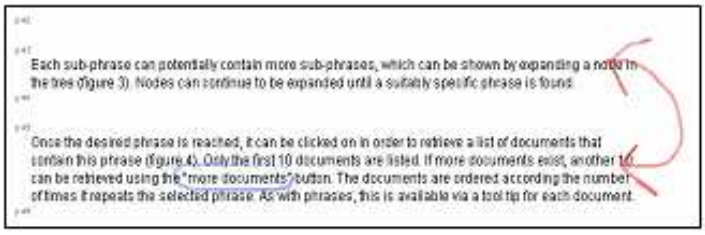


Figure 5.7: Free hand drawing

Both types of gestures are always painted in the unique colour that relates to the user who made the gesture; thus maintaining ownership awareness. Also common to both types of gestures is the fact that they are not persistent and gradually fade away. This reflects the fact that gestures are temporary and therefore quickly loose their usefulness as the document around them changes and the context of what is said or done at the time of the gesture is forgotten. However, since the gestures are short-lived, they can only be played out to those users who are in the same location within the document as the user performing the gestures. When a gesture is received by a user who is not in a position to see it, an indicator appears to the right of the scrollbar showing the location and owner of the gesture (Figure 5.8). Clicking on this indicator changes the user's view to the location of the gesture, and then the gesture is played back as it was drawn. These indicators, like the gestures themselves, gradually fade over time so that they can be ignored if they are of no interest to the user. If the user decides however to click on this indicator, he will be taken to that position in the document and will see the gesture being played. In this case, an anchor is displayed on the user's scrollbar so he can conveniently return to his original location.

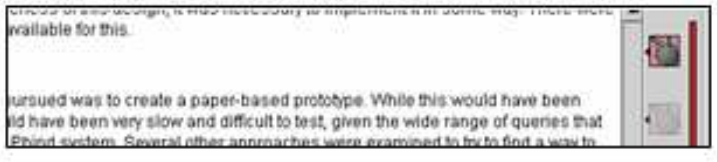


Figure 5.8: Gesture indicators

Lock Status Awareness

RECOLED highlights the effects of the locking mechanism by transparently showing the mechanism itself in the interface. This is achieved by using the regions to the left of each paragraph (Figure 5.4.) which change colour to indicate the status of the paragraph lock. White indicates that the paragraph is unlocked, and the user is free to edit it. Grey indicates that a lock has been requested but has not yet been confirmed, which means that there is a chance that any modifications to this paragraph will be ignored. Any other colour indicates that the paragraph is exclusively locked by someone else who is represented by that colour. A locked paragraph cannot be edited by anyone other than the current owner. However, the lock is automatically released if the owner is not active for a specific period of time (currently set at 3 seconds of inaction).

5.4 Interaction Metadata Generation

One of our design assumptions is that when writing text, co-authors often naturally use text segmentation to structure their document into semantic units. Our aim is to attempt to keep track of these semantic units from their creation, and follow their evolution during the co-writing process.

In RECOLED, it is assumed that the appropriate granularity of the text unit for this purpose is the paragraph; defined as a segment of text followed by a blank line (*"text\n"*). Metadata generated is stored with the document content in XML format for convenient post-meeting processing.

5.4.1 Editing Primitives

RECOLED is primarily designed to accurately capture editing operations. The original set of editing operation primitives defined was designed to be as general as possible, and included:

Insertion: insertion of a single character

Deletion: deletion of a single character

Gesturing: use of a tele-pointer or other deictic widgets used on one or more text segments.

However, trials of earlier prototypes quickly showed that these were clearly insufficient. In particular, insertions or deletions of a new line are highly significant operations in that they modify the structure of the document. To address these particular cases, two new primitives were added to the previous set:

Paragraph Insertion: insertion of a new paragraph (i.e. a newline : "\n")

Paragraph Deletion: deletion of an existing paragraph

Finally, someone using a single editor would naturally expect to be able to cut and paste text. These operations, supported by RECOLED, are also significant because once again, their use can possibly modify the structure of the document. As such, two further primitives were also defined:

Paste: insertion of strictly more than on character in a single editing operation

Cut: deletion of strictly more than on character in a single editing operation

5.4.2 Timestamping

The timestamping mechanism used in RECOLED generates two distinct sets of timestamps. The first set consists of paragraph timestamps, as illustrated in Figure 5.9. whose purpose is to describe the nature of editing operations performed on every single paragraph. These contain information on the agent who performed the operation, the type of action (i.e. is one of the editing primitives previously described) the start and end time of the action, as well as a reference to a unique action timestamp. In the XML representation of the document, text segments consist of the paragraph content and the paragraph timestamps which were generated on that particular paragraph. If the document is structurally modified, RECOLED ensures that these paragraph timestamps are handled accordingly. We later describe, and illustrate it with a few examples, the design choices that were made to handle these timestamps.

```
<segment id="4.1">
  <timestamp actionid="17" agent="2" action="NewLine_Insert" start="215" end="215"/>
  <timestamp actionid="19" agent="2" action="Insert" start="215" end="217"/>
  <timestamp actionid="20" agent="2" action="Delete" start="220" end="222"/>
  <timestamp actionid="21" agent="2" action="Insert" start="221" end="221"/>
  <timestamp actionid="22" agent="2" action="Insert" start="222" end="226"/>
  <timestamp actionid="24" agent="1" action="Insert" start="231" end="231"/>
  <timestamp actionid="57" agent="2" action="Insert" start="486" end="495"/>
  budget of 3000 from the student union
</segment>
<segment id="4.1.1">
  <timestamp actionid="58" agent="2" action="Insert" start="496" end="498"/>
  <timestamp actionid="59" agent="2" action="Delete" start="498" end="499"/>
  <timestamp actionid="60" agent="2" action="Insert" start="499" end="502"/>
  <timestamp actionid="61" agent="2" action="Delete" start="503" end="503"/>
  <timestamp actionid="62" agent="2" action="Insert" start="504" end="505"/>
  maybe charge people more?
```

Figure 5.9: RECOLED paragraph timestamps

The second set of timestamps consists of action timestamps, as illustrated in Figure 5.10. Action timestamps accurately describe editing or gesturing operations performed on the document. The information they contain are: a unique identifier, the type of operation performed, the start and end time of the operation, the list of paragraphs which were affected by this operation, the offset (within the first paragraph affected by the operation) at which the operation started, and in the case of gesturing the positions of the gesturing. In addition, they contain the exact text content of the operation performed (e.g. what text was added or deleted), and in the case of gesturing the text which was pointed at. Even though there is a certain redundancy in the information contained in these timestamps, there is a subtle difference in their nature. For example, a single action timestamp would describe the operation of pasting simultaneously 3 new paragraphs on the document. This same operation, on the other hand, will generate 3 distinct paragraph timestamps, one for each of the paragraphs affected.

5.4.3 Timestamping Manipulation Rules

The final version of a document text follows a linear path from start to end. Such document can be represented by the familiar tree structure of Figure 5.11a. This, however, fails to describe the often laborious process by which the text was produced. For instance, words might have been written or deleted in arbitrary locations, or paragraphs might have been merged, split and moved in many different ways. Although representing the structure of a text at a given state is a straightforward operation, attempting to make a structured representation of a document as the sums of its editing operations can quickly become extremely complex and intractable. In RECOLED, the actions timestamps describe completely the evolution of the document, and applying these actions at their time of appearance would in effect replay the process by which the

```
<action id="50" type="Insert" startT="428" endT="433" paragraphs="6.1.1" startOffset="0"> 4400 </action>
<action id="51" type="Insert" startT="437" endT="437" paragraphs="6.1.1" startOffset="0"> $ </action>
<action id="52" type="Delete" startT="438" endT="440" paragraphs="6.1.1" startOffset="0" endOffset="1"> $4 </action>
<action id="53" type="Insert" startT="438" endT="452" paragraphs="6.3" startOffset="1"> 2700 for staff in single room </action>
<action id="54" type="Insert" startT="439" endT="439" paragraphs="6.1.1" startOffset="0"> 4 </action>
<action id="55" type="Insert" startT="442" endT="448" paragraphs="6.1.1" startOffset="4"> for students11 </action>
<action id="56" type="Delete" startT="449" endT="450" paragraphs="6.1.1" startOffset="17" endOffset="18"> 11 </action>
<action id="57" type="Insert" startT="486" endT="495" paragraphs="4.1,4.1.1" startOffset="15"> from the student union </action>
<action id="58" type="Insert" startT="496" endT="498" paragraphs="4.1.1" startOffset="0"> maybe chga </action>
<action id="59" type="Delete" startT="498" endT="499" paragraphs="4.1.1" startOffset="8" endOffset="9"> ga </action>
<action id="60" type="Insert" startT="499" endT="502" paragraphs="4.1.1" startOffset="8"> arge people </action>
<action id="61" type="Delete" startT="503" endT="503" paragraphs="4.1.1" startOffset="18" endOffset="19"> e </action>
<action id="62" type="Insert" startT="504" endT="505" paragraphs="4.1.1" startOffset="18"> e more? </action>
<action id="63" type="Insert" startT="568" endT="573" paragraphs="7.1.1" startOffset="0"> 24*220 = </action>
<action id="64" type="Insert" startT="590" endT="596" paragraphs="7.1.1,7.1.2" startOffset="9"> 5280? </action>
<action id="65" type="Insert" startT="598" endT="598" paragraphs="7.1.2" startOffset="0"> + </action>
<action id="66" type="Insert" startT="609" endT="615" paragraphs="7.2" startOffset="0"> 27*220=2 </action>
```

Figure 5.10: RECOLED action timestamps

document was created. The paragraph timestamps, on the other hand, attempt to describe as accurately as possible the evolution of the text segments. As such, some paragraph timestamps might no longer be reflected in the actual text content, and instead might describe an operation which was performed when the paragraph was in an earlier state. In what follows, we describe issues which arise when attempting to maintaining a log of actions on textual entities (paragraphs) when the meeting document is subject to structural modifications (*e.g. moving, merging, splitting textual entities*) during the course of a meeting and describe the manipulation rules we propose in order to resolve these manipulation issues (Bouamrane et al. 2005).

Moving Paragraphs

Figure 5.11 illustrates the evolution of the structure of a plain style text document, when its first paragraph was removed and its content was appended to the end of the document. The paragraphs "Par" represent positions within the document structure and make no reference to textual content. The actual text content is represented by the"text\n" boxes. Figure 5.11a. shows the original state of document while Figure 5.11b. displays the result after the segment *"text1\n"* has been cut and appended.

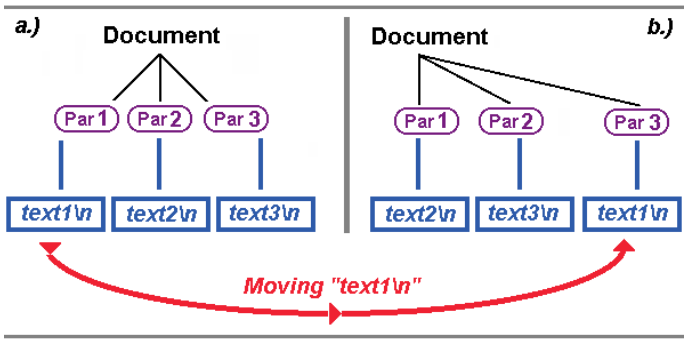


Figure 5.11: Moving the textual content *"text1\n"* from first to third paragraph position in the document structure

In this scenario, the list of timestamps which described the operations performed on the content of *"text1\n"* prior to its move will follow the paragraph to its new position (illustrated in Figure 5.12). In order to achieve this, the paragraph timestamps lists are mapped to the paragraphs' contents. Action timestamps will record both the paste and cut operations, and in addition, in the paragraph pasted, a newly generated "paste" timestamp will be appended to the previous list of paragraph timestamps.

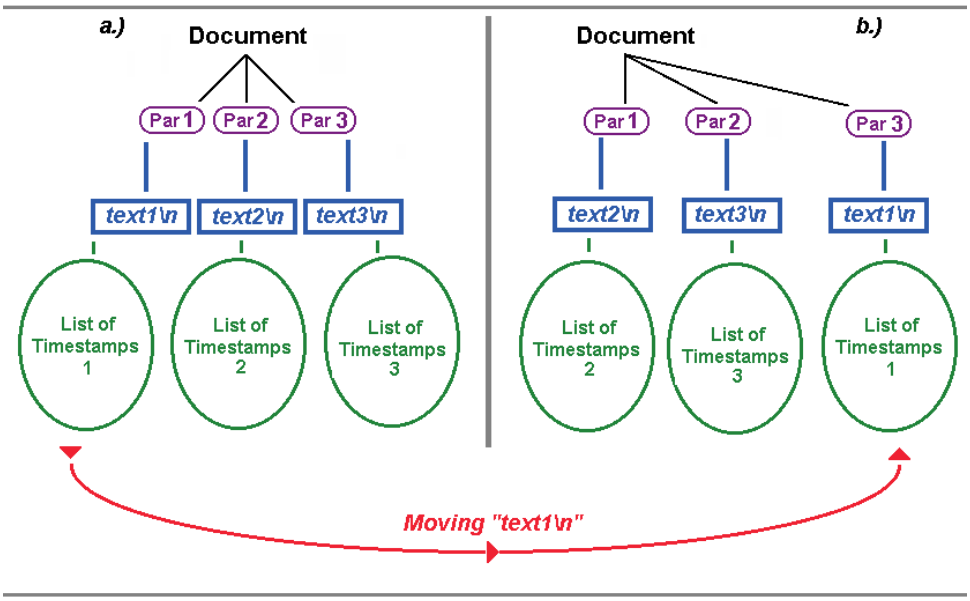


Figure 5.12: Effect of paragraph content manipulation on paragraph timestamps

Merging paragraphs

Steps similar to those taken when a paragraph is moved are also taken when paragraphs are merged. In Figure 5.12, the deletion of the new line at the end of *"text3\n"* will merge paragraphs 2 and 3 (illustrated in Figure 5.13). In this case, an action timestamp will record the operation as a "Paragraph Merge". However, as paragraph 3 disappears, its list of timestamps is appended

to that of paragraph 2. In this context, the timestamps which recorded operations on the content of *"text1\n"* are not necessarily relevant to the content of *"text3\n"*. This however models the fact that paragraph 2 is a merge of two separate text segments, and therefore, its timestamps describe operations which were performed on earlier states of these two separate text segments and eventually led to the current state of paragraph 2. Since such manipulations only happen on the paragraph timestamps, and as action timestamps never change, possible ambiguities can always be resolved by the combinative use of the two sets of timestamps. Action timestamps will determine exactly when a merge happened.

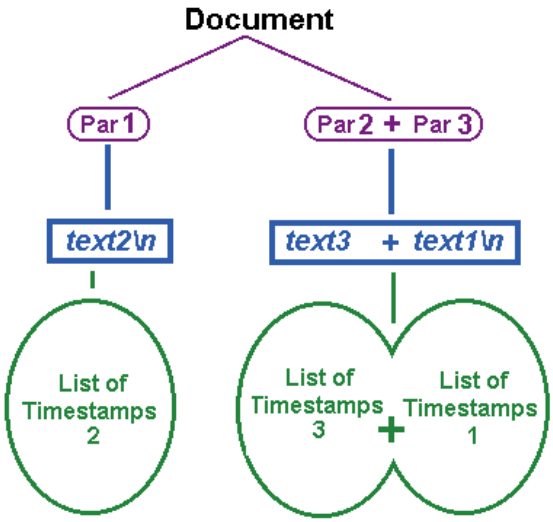


Figure 5.13: Effect of merging several paragraphs on paragraph timestamps

5.4.4 Examples of timestamps manipulation

The timestamping model which has just been outlined is useful for describing "clean cut" operations, where paragraphs are manipulated as atomic units. However, our experience from using RECOLED in co-authoring situations shows that text operations are often rather complex, and therefore require a more advanced timestamping model. In what follows we will describe our design choices based on a number of practical situations arising when writing a document, and propose a taxonomy of cut operations and their effects on paragraph timestamps.

5.4.5 Paragraph naming scheme

RECOLED assigns a unique identifier to each paragraph to allow a mapping of timestamps to paragraphs. The paragraph identifiers correspond to their relative positions within the document structure. When a paragraph is inserted between two existing paragraphs, these paragraphs are not renamed, but instead, the new paragraph is assigned an id by assigning an extra decimal point to the id of its preceding paragraph, as illustrated in Figure 5.14.

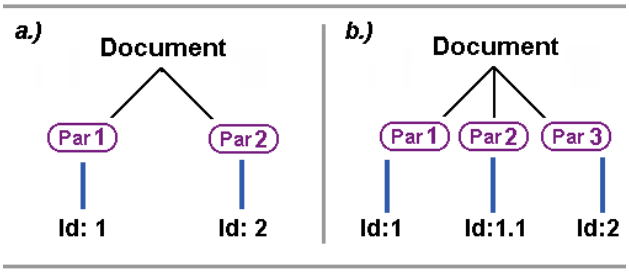


Figure 5.14: RECOLED paragraph naming scheme

5.4.6 Paragraph Split

If a new line is inserted in the middle of an existing paragraph, this will split the paragraph in two, which we can call paragraph 1 (first part) and paragraph 2 (second part). One possible solution would be to duplicate the list of timestamps into both paragraphs: this is akin to saying that there are now two new entities derived from the previous paragraph and that they both "inherit" the history log of their parent (Figure 5.15a.). Another solution would be to assign the history to the first half of the paragraph and start a new log for the second half: this is akin to saying that the first half of the entity is really the same entity as the original one, and that the second one is a completely new entity (thus with a new history log) (Figure 5.15b.). Finally, the third option would be to examine the existing timestamps of the entity and to split them according to the content of the two new entities (perhaps though string matching). Given the constraints of resolving paragraph timestamp manipulations in real-time, the third solution was impractical while neither the other two options were entirely satisfactory. In the end a compromise was found in considering a paragraph split to be a new entity (and thus having a new log) if its size was less than a certain percentage (25%) of the original paragraph. If both parts are above this threshold, the original history log gets replicated. This solution was found to be a reasonable compromise in the face of real-time constraints.

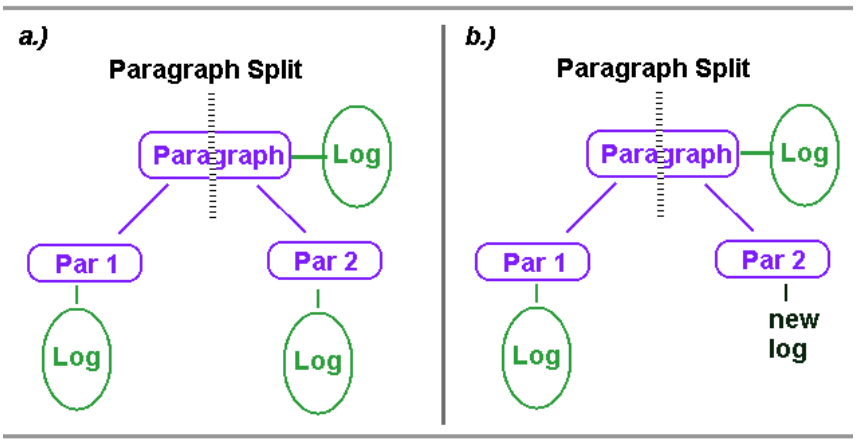


Figure 5.15: Two possible timestamping manipulation rules in the "Paragraph Split" case

Table 5.1: Taxonomy of Cut operations and their effect on paragraph timestamps

Title	**Beginning Cut**
Description	(startC < startP) and (startP <= endC < endP)
Example	15 staff, **how many students?** **40 students** but some have their...
Effect on Document Structure	Paragraph is merged with previous paragraph
Action taken on paragraph timestamps	Timestamps are merged with previous paragraph timestamps
Title	**Middle Cut**
Description	(startC > startP) and (endC <= endP) or (startC >= startP) and (endC <endP)
Example	15 staff, how many students? 40 students **but some have** their...
Effect on Document Structure	No change to paragraph in document structure
Action taken on paragraph timestamps	No action taken on paragraph timestamps
Title	**Whole Cut**
Description	(startC <= startP) and (endC >= endP)
Example	15 staff, how many students? **40 students but some have their...**
Effect on Document Structure	Paragraph disappears
Action taken on paragraph timestamps	paragraph timestamps are mapped to paragraph content, and later retrieved if paragraph is subsequently pasted
Title	**End Cut**
Description	(startP < startC < endP) and (endC >= endP)
Example	15 staff, **how many students?** **40 students** but some have their...
Effect on Document Structure	No change to paragraph in document structure
Action taken on paragraph timestamps	No action taken on paragraph timestamps

5.4.7 Taxonomy of Cut operations

We have identified a number of different situations in which a paragraph can be affected by a "Cut" operation. These are represented in Table 5.1. They are identified by the position (start and end offsets) of the paragraph relative to the cut operation, along with their effect on document structure, the action taken regarding the paragraph timestamps, and an example to illustrate each case. In the table, startP and endP correspond to the start and end offset of a paragraph and startC and endC correspond to the start and end offset of a Cut operation. In this example, the cut corresponds to the text in bold and the portion of the paragraph described by the cut operation taxonomy is underlined. The other paragraph is there to illustrate the relative position in the document. All cut operations are combination of the cases described in Table 5.1. The examples described in Table 5.2 shows how a single cut operation (represented by the text in bold) can be broken down into three types of cut.

Table 5.2: Break down of a single cut operation into the types defined in Table 5.1

Par_1: 15 staff, **how many students?** Par_2: **40 students but some have their own travel arrangements** Par_3: **10 students coming by car**, need to organise transport for 45 Par_4: A bus for up to 60 cost 1500	
Cut types:	Par_1: *"end cut"* Par_2: *"whole cut"* Par_3: *"beginning cut"*

Middle cut and end cut do not affect the paragraph in the document structure, so the original paragraph retains its list of timestamps. If the chunk of paragraph cut is subsequently pasted, the newly pasted paragraph will inherit the previous list of paragraph timestamps, according to the timestamps manipulation rules previously described in this section.

5.5 Summary

We have presented RECOLED, an integrated real-time multimedia meeting architecture that integrates traditional editing and awareness functionality with mechanisms designed to implicitly capture document history, based on the Artefact History Model detailed in Chapter 4. We presented a model for automatic interaction history generation based on the manipulation of textual items (a paragraph and its' interaction history), we defined editing and gesturing action primitives, and proposed rules for the manipulations of paragraph logs and illustrated these rules with examples taken from real editing scenarios.

Chapter 6

RECOLED Usability Study and Meeting Corpus Collection

6.1 Introduction

In this chapter, we present a usability study of the RECOLED remote meeting environment, followed by the description of our meeting corpus collection. The reason why these two topics are covered in the same chapter is due to the fact that both activities were carried out in parallel: we captured the meetings held by the participants of the RECOLED usability study and these were subsequently used as the core of our meeting corpus. The description of the meeting corpus was published in Luz et al. (2006).

6.2 Usability Study of RECOLED

A usability study of RECOLED was conducted to evaluate its capabilities as a synchronous shared editor for supporting collaborative writing (Masoodian et al. 2005). This study involved two separate sets of experiments. The methodology used in both experiments was identical but the tasks that participants had to perform were significantly different. This was done in order to see how different scenarios might impact on the use and perceived strengths or weaknesses of the various features of RECOLED. The aim of this study was not only to evaluate the group awareness and other features of RECOLED, but also to collect meeting data which could be used for post-meeting analysis of the collaborative meeting contents.

6.2.1 Methodology

The study was conducted at two usability labs (University of Waikato and Dublin, Trinity College). In the first experiment, carried out by David King in Waikato University, 14 students took part

in seven dyadic writing sessions, and received a book voucher for taking part in the study. In the second experiment, carried out by the author in Trinity College Dublin, 24 students were involved in 12 dyadic sessions. Participants were usually paired with someone of their choice prior to arranging the experiment. The 14 participants in the experiment in Waikato University were final year computer science students. The 24 participants in the experiment in Trinity College Dublin were postgraduate computer science students from a variety of research groups (Computer Linguistics, Artificial Intelligence, Machine Learning, Interaction Systems and Distributed Systems groups). 10 participants were female and the rest were male. There were 2 all female pairs, 6 mixed pairs and the remaining 11 all male pairs. All the participants were familiar with the use of text editors, but had not previously used a shared editor. During the experiment itself, the participants were located in separate rooms and could not see each other so all communication was carried out through the use of the RECOLED shared-editor and the complementing RAT (University College London's Robust Audio Tool). There were only ever one pair of participants in the usability lab for each experiment.

Editing tasks

The group task used for the first study (Task A) required the subjects to collaboratively work on a simple children's story in which the paragraphs had randomly been rearranged and some of its words were replaced with underscores. The participants' task was to cooperatively arrange these fragmented paragraphs in a logical order and to replace the missing words to recreate the story. The task was specifically designed to encourage a high level of interaction and communication between the participants. Participants in the second study were asked to organize a fictional week-end social function for final year students and staff (Task B). The things they were asked to organize included booking a hotel, travel arrangements, proposing day time and night time activities, assessing costs, etc. Upon starting the session, a basic plan was loaded on to the editor with a number of outstanding issues to be resolved. In order to do this, participants were each handed a (small) set of printed information about the destination's accommodation availability and attractions. Participants had access to both shared and exclusive information. This was specifically designed to encourage interaction and communication between the participants and to ensure that for certain tasks, each person had to rely on the other participant's critical information.

Sessions

Each session began with a demonstration of the software, which typically lasted half-an-hour to 45 minutes. The participants were then encouraged to explore the shared-editor functionality and to ask any questions they might encounter. In addition to RECOLED, participants could communicate via an audio channel through the RAT audio conferencing tool, which required the subjects to wear a headset. Once the participants indicated that they were confident in using

the program, the experiment proceeded to the actual evaluation task. There was no time limit set for completing the group task, and the participants were free to continue on the task for as long as they wished. Once the participants felt that they had completed the task satisfactorily, they were asked to fill in a short questionnaire. This was followed by an informal discussion with the researcher, which explored the participants' views about the software, the task, collaborative writing in general, and specific suggestions for improvements to the system. The sessions usually took about an hour to an hour and a half each.

Data recording

The document history capture was done by the automatic editing-gesturing timestamps generation mechanism of RECOLED, detailed in the previous chapter. Audio recording of the sessions were made through the RTP-recording functionality of the RECOLED server.

6.2.2 Usability Results and Implications

The use of questionnaires, informal discussions, and general observations made during the sessions helped us gain insight into the participants' view of the software and its features, as well as its overall usability. The main findings of the two studies are discussed below.

Writing: The participants adopted a wide range of writing strategies, which were different not only across the two tasks, but also across different sessions with the same task. This is consistent with the findings of other studies which show that people use a variety of approaches to collaborative writing (Posner and Baecker 1992). In our studies the writing methods also tended to change within each session as participants previously unfamiliar with a co-authoring task, gradually became more comfortable with using the shared editor. An interesting point was to see how quickly participants accepted the idea of sharing a document. As documents were loaded at the start of the sessions, the co-authors made no ownership claims on the document and had no objections in the other participant modifying it in any way. In the first study the most common strategy was for participants to act as joint writers, working closely together and consulting each other carefully before taking action. This demonstrated the effectiveness of RECOLED as a shared writing tool. Sometimes, however, the participants worked as separate writers, taking responsibility for different tasks or sections of the document and working independently of each other. A common strategy was to have one participant filling in the blank words while the other searched for paragraphs and rearranged them. In the second study the participants generally worked in tight cooperation, tackling a specific task before moving on to the next one. An interesting point was to see how the various groups perceived priorities differently and therefore chose to address the problems in different orders. Although a basic plan was loaded on to the editor when the session started, the

participants did not address the plan items in the order in which they appeared, nor did they try to physically reorder these items within the document. Different groups did however address the items in different orders, according to their own views of priorities (cost, travel arrangements, etc.). It would be impossible to infer these differences by solely looking at the final document. Therefore, the users' perceptions of priorities would have been lost if the planning was recorded using a traditional shared editor. One of our assumptions is that by capturing the time and type of user actions, post-meeting processing of timestamps generated by RECOLED will permit to emphasize different groups' sense of priorities when working on a similar task.

Gesturing: Gesturing was used extensively during the writing process in Task A, although mainly in the form of pointing rather than freehand drawing. Also, gesturing was always accompanied by audio conversations and never on its own. Once again, the fact that some form of gesturing was used extensively, and that it was always in conjunction with audio conversations, demonstrates the value of recording both the gestures and audio in document history, as a way of identifying co-authors' focus during the collaborative writing process. Gesturing was used very little during Task B. A possible reason for this is that since the planning document was rather short, the participants were always seeing the same thing on their screens. The tight cooperation involved in this scenario, coupled with the availability of the audio channel made the focus of the co-authoring task very clear, thus requiring little or no additional gesturing.

Highlighting: A surprising result was that highlighting was also used frequently during the first study, even though the writing process was synchronous. In the case of this study highlighting was generally used to mark what words were added in place of the missing words. This simple but effective use of highlighting shows that mechanisms such as highlighting are an integral part of co-authoring, and information gained from recording its use can be valuable.

Shared scrollbars: Shared scrollbars were also used heavily during the first study. The participants not only used shared scrollbars for locating the other participant's view, but also on many occasions linked their views to follow one another as they moved through the document. This feature, however, caused some disorientation when one of the participants would for instance forget that their views were linked, and the other participant would scroll through the document.

Text locking: Another interesting result is that the participant almost never edited a single paragraph together during either of the studies. An obvious reason for this is the paragraph-based locking mechanism of RECOLED, which was demonstrated to the users before the sessions began. Therefore, the participants knew that they could not edit the same paragraph at the same time and rarely tried to do so. Also, when an audio channel is available, people

tend to discuss who should be working on what part of the document verbally. Therefore, the simple optimistic locking mechanism used by RECOLED seems to be sufficient in preserving documents consistency without hindering the co-authoring task.

6.2.3 Participants' feedback

After the completion of the sessions, participants were handed a questionnaire divided into four sets of questions relating to the task performed, their view of their collaboration, the usability of the shared editor and the use of the audio channel.

Collaboration: Participants felt comfortable about sharing the document. An interesting point is that when asked to rate their own and their partner's contributions to the overall task, the participants felt that they had contributed equally to the task. They also generally thought that they were rather efficient in completing the required tasks and were satisfied with the final outcome.

Shared-editor general usability: The participants generally found the awareness widgets to be very useful. As to what could improve the editor, a general feeling was that it would be useful to be able to import other types of objects such as pictures and tables. An encouraging point is that many felt they could easily see themselves using the tool for a number of tasks such as: planning tasks, project and budget proposals, document review, editing papers, computer programming, or as instant messaging or chat tool.

Audio Communication: The ability to speak with each other was usually rated very highly. This is consistent with previous findings (Jensen et al. 2000, Bos et al. 2002) regarding the importance of audio in building trust between participants and encouraging cooperation. As an audio channel was part of the experiment set-up, many participants just took it for granted and assumed that it would not be possible to complete the task without it.

6.3 Corpus Collection

As we have seen in Chapter 2, the study of multimodal meetings currently attracts considerable interest among a wide variety of research communities. Despite this high level of interest, corpora of multimodal meetings are not easily available and many researchers have developed their own corpora, tailored to their particular research interest. The lack of publicly available meeting corpus has recently in part been addressed by the release of the AMI (Augmented Multiparty Interaction) corpus (*AMI Project* 2006). This corpus consists of a 100 hours of co-located, multi-modal meeting recordings, involving a small number of people in a variety of scenarios (Carletta et al. 2005, AMI Corpus 2006). The fact that such a significant multi-modal meeting corpus should be made freely available to the research community will without doubt result in important progress being made

in the field of meeting browsing and particularly regarding issues relating to meeting browsers' evaluation methodologies and benchmarks (see Chapter 9 for more details on this topic).

The corpus described in this section was collected in order to meet our specific research interests: namely, meetings recordings associated with detailed interaction metadata. Unlike existing meeting corpora, the corpus described in this section emphasises temporal relationships between speech and text media streams(Luz et al. 2006). This is achieved through detailed logging and timestamping of text editing operations, actions on shared user interface widgets and gesturing, as well as generation of speech activity profiles, as described in Chapter 5. However, the corpus and tools used in collecting it were designed so that wide range of phenomena could also be investigated in detail.

6.3.1 Meeting Scenarios

The meeting scenario we have targeted is one where a group of collaborators (typically two) synchronously write a text document which reflects the results of an oral discussion held simultaneously with the collaborative writing activity. Examples of such scenarios include collaborative writing of minutes, joint preparation of articles, work plans etc.

6.3.2 Meeting Data

The overall meeting and data collection architecture is shown in Figure 5.2. The data collection tools and library, including RECOLED (meeting environment), RECPLAJ (recording server) and REXPLORE (reference library) have been released as free software. Further information can be found in the project's web site (COWRAT 2006).

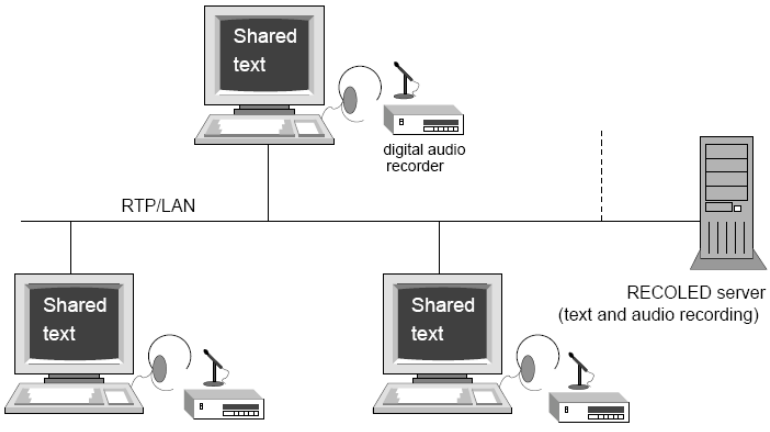


Figure 6.1: Meeting Recording Architecture

Annotation of the recorded data contain the following metadata elements:

- basic *segmentation* to establish text and speech units,

- *time stamps* to keep track of actions (write, delete, copy etc) performed by each participant on each text segment of the resulting text,
- detailed *action descriptions*, including actions performed on segments deleted from the final text, and
- user-defined *keywords* with which participants can highlight text they consider relevant.

These metadata elements are encoded in XML. Figure 6.2 shows a simplified version of the document type definition (DTD) used for text encoding. Meeting participants also communicate through speech with the aid of a multicast audio tool. Audio communication is mediated through the Real Time Protocol (RTP). A server (RECPLAJ) records RTP data and control packets exchanged during the meeting. Recorded RTP control packets contain reports of reception quality as well as various conferencing events. RTP data packets include packet header information such as synchronisation source (participant) identifiers and contributors (in the case of mixed-source packets), sampling time stamps, packet sequence number and payload type. Speech profiles are derived directly from such recorded packets by monitoring of channel activity (packet arrival rate). A speech profile is extracted for each participant. Audio profiling at the moment serves the single purpose of distinguishing silence and speech-filled time intervals. Audio profiles are not directly connected to text metadata.

The audio format used for transmission over RTP is GSM encoding at a sampling rate of 8 KHz. Although this setting ensures reasonable audio quality over the network, the resulting recordings are inadequate for automatic speech recognition. Therefore, we have recently begun to complement RTP recording with local recording of high quality audio. Each participant is provided with a portable digital recorder and clip microphone. Local recordings use a sampling rate of 44Khz and a bit rate of 320 Kbit/s and are later synchronised with the system's audio and text tracks.

6.3.3 Corpus Description

A corpus of thirty-three meetings has been collected to date. These meetings have been organised into four distinct sets according to the type of tasks the participants were asked to perform:

(A) reordering an existing text,

(B) organising a weekend break,

(C) discussing a research project,

(D) miscellaneous meetings.

Set A and B were dyadic writing sessions organised as part of the usability study of the collaborative writing environment described in the first part of this chapter. In set A, a total of

```
<!ELEMENT comapdoc (meeting|section|segment|actions)*>

<!-- Meeting metadata: venue, description etc -->
<!ELEMENT meeting (venue,description,partlist)>
<!ATTLIST meeting date CDATA #REQUIRED>
<!ELEMENT venue (#PCDATA)>
<!ELEMENT description (#PCDATA)>
<!ELEMENT partlist (participant+)>
<!ELEMENT participant (#PCDATA)>
<!ATTLIST participant id CDATA #REQUIRED>

<!-- Meeting data: some basic structure -->
<!ELEMENT header (#PCDATA | timestamp)*>
<!ATTLIST header
          level CDATA #IMPLIED>
<!ELEMENT segment (#PCDATA|keyword|header|timestamp)*>
<!ATTLIST segment
          id CDATA #REQUIRED>
<!ELEMENT section (segment+)>
<!ATTLIST section
          level CDATA #IMPLIED>

<!-- timestamp tag; It can be further constrained -->
<!-- so that actionids match action id values -->
<!ELEMENT timestamp EMPTY>
<!ATTLIST timestamp agent CDATA #IMPLIED
                    action CDATA #IMPLIED
                    actionid CDATA #REQUIRED
                    start CDATA #REQUIRED
                    end CDATA #REQUIRED>

<!ELEMENT actions (action*)>
<!ELEMENT action (#PCDATA)>
<!ATTLIST action id CDATA #REQUIRED
                 type CDATA #REQUIRED
                 startT CDATA #REQUIRED
                 endT CDATA #REQUIRED
                 points CDATA #IMPLIED>

<!ELEMENT keywordTypes (keywordType*)>
<!ELEMENT keywordType #EMPTY>
<!ATTLIST keywordType id CDATA #REQUIRED
                      name CDATA #REQUIRED>
<!ELEMENT keyword #EMPTY>
<!ATTLIST keywordType id NMTOKEN #REQUIRED
```

Figure 6.2: Document type definition for collaborative meeting document

				text actions		gesturing actions	
Task	no. of meetings	total duration	avg. text length	total	average	total	average
A	7	295 min.	6635 words	1608	229.7	633	90.4
B	9	224 min.	3918 words	1698	188.7	89	9.9
C	9	412 min.	2690 words	1165	129.4	103	11.4
D	8	212 min.	1701 words	1422	136.0	300	75.0
total	33	19 h. 03min.	-	5893	-	1125	-

Table 6.1: Corpus composition according to task

fourteen students from the department of computer science of the University of Waikato took part in seven dyadic writing sessions, and received a book voucher each for taking part in the study. The group task required the subjects to collaboratively work on a simple children's story in which the paragraphs were randomly rearranged and some of its words were replaced with underscores. The task essentially consisted of cooperatively arranging these fragmented paragraphs in a logical order and to adding in the missing words to recreate the story. The task was specifically designed to encourage a high level of interaction.

Set B contains recordings of meetings in which participants were asked to organize a (fictional) weekend social function for final year university students and staff. This scenario included a number of sub-tasks such as booking a hotel, organising travel arrangements, proposing daytime and night-time activities, assessing costs, etc. Sets of handouts containing information relevant to the task were distributed to the participants before the meeting. During the meeting, participants had access to shared and exclusive information. This task was specifically designed to encourage interaction and communication between the participants and to ensure that for certain tasks, each person had to rely on the other participant's critical information. A total of twenty four participants performed this task in dyadic sessions. The majority were postgraduate computer science students, from Trinity College, Ireland, but a few first year undergraduate students also took part in this experiment.

Set (C) consists of recordings of student-supervisor meetings relating to ongoing final (fourth) year students projects in the Department of Computer Science at Trinity College. These recordings aimed at exploring interaction in a real-life collaborative situation and keeping track of the evolution of the collaborative process as users become more accustomed to synchronous remote collaboration in general, and the RECOLED tool in particular.

The remaining set (D) consists of miscellaneous meetings with a variety of topics, including organising a table quiz, making arrangements for a bank holiday week-end, organising a concert and organising a travel itinerary for a group travel in China. The last 2 meetings (concert & itinerary) were later picked for a task-oriented evaluation test of a meeting browsing tool, further described in Chapter 9.

6.4 Summary

In this chapter, we have presented the results of the usability study of the RECOLED meeting environment as well as the details of the collection of a meeting corpus, which emphasises temporal relationships between speech and participants' interaction streams. We found that overall the meeting architecture combining shared-editing and audio communication enabled effective collaboration on a variety of tasks. The RECOLED shared-editor exhibited good usability and awareness features and encouragingly, a majority of participants found that the tool could be useful for col-

laborating on common tasks. In the following chapter, we will see how interaction information can be used as the basis for providing novel navigation modalities in meeting recordings and we present a meeting browsing tool primarily based on meeting interaction information. The corpus collected was used extensively to test the performance of this meeting browsing tool in an analytical evaluation detailed in Chapter 8 and usability evaluation described in Chapter 9.

Chapter 7

An Interaction-based meeting browsing tool: the Meeting Miner

7.1 Introduction

In this chapter, we present various ways in which the interaction metadata collected in real-time by the RECOLED meeting architecture can be used for navigating meeting recordings. We first present the underlying meeting representation, semantic models and search concepts which will be used as the basis for providing meeting information access. We will then demonstrate how interaction information can be used as a novel meeting navigation modality, namely in the form of interaction-based navigation and paragraph temporal neighbourhood indexing. These navigation paradigms are subsequently implemented in the Meeting Miner, an interaction-based meeting browsing system. We describe its various functionalities and the user interface in details. While the design behind the meeting browsing tool will occasionally seem to rely on empirical heuristics, every effort was made to ensure these could be justified with sound information modelling (Bouamrane and Luz 2007a). In addition, the analytic evaluation described in the following Chapter 8 will provide quantitative results to measure the practical validity of these heuristics, while the usability study described in Chapter 9 illustrates how users performed in practical tasks with the Meeting Miner's functionalities . Finally, we describe how we incorporate an ASR component into the browsing system as a complement to interaction-based navigation. We conclude this chapter with a few concrete examples which serve to illustrate meeting navigation scenarios with the Meeting Miner browsing tool (Bouamrane and Luz 2006a).

7.2 Meeting Modelling

7.2.1 Meeting Representation

The unstructured meeting data at our disposal for each meeting of our meeting corpus now consist of: (i) the meeting textual outcome $\mathcal{O}$, (ii) individual audio recording of the meeting participants, (iii) the speech profiles of individual participants (these are automatically inferred from RTP packet count) and finally, (iv) meeting interaction metadata. This combination of data can lead to a variety of possible meeting representations, and several of these are illustrated in Figure 7.1.

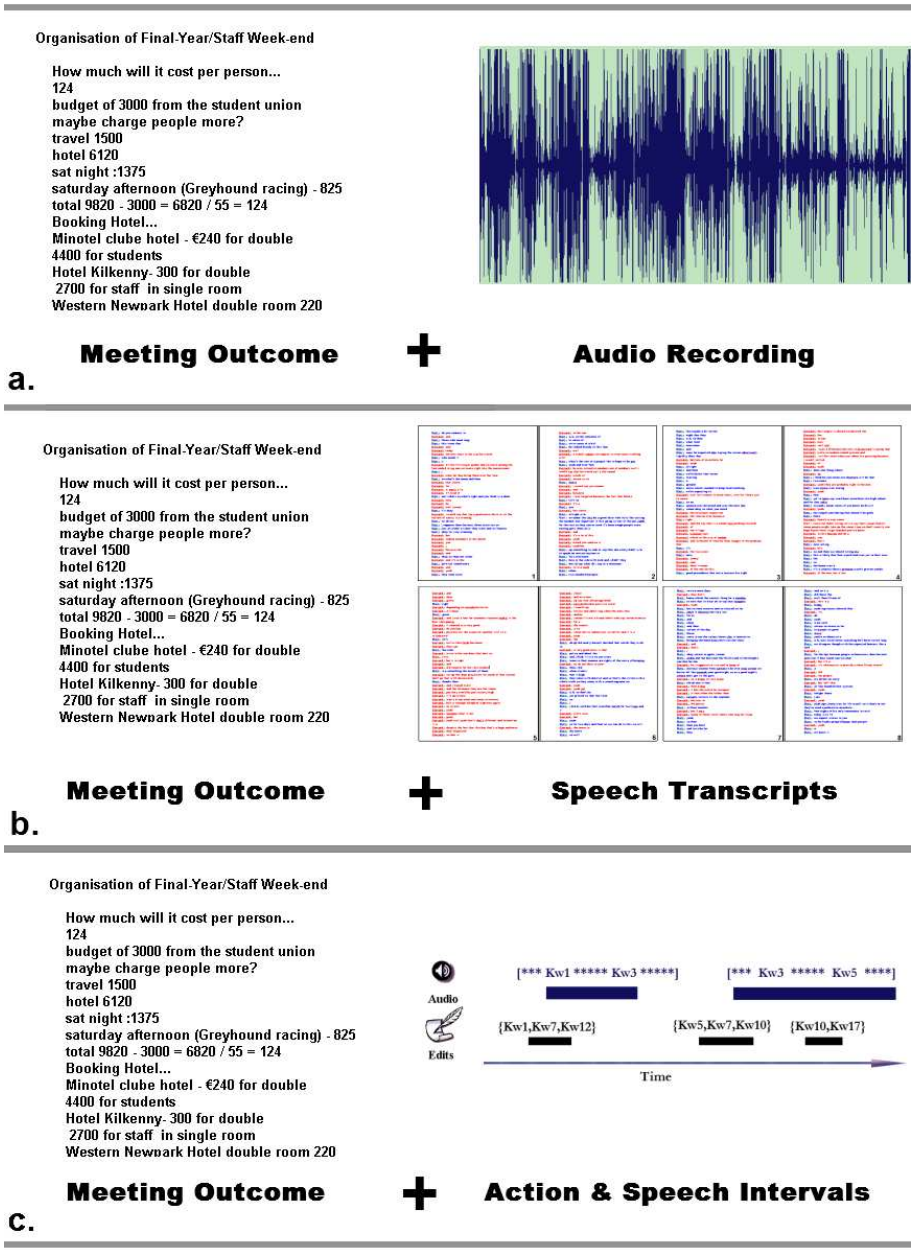


Figure 7.1: Three possible Meeting Representations

Raw Meeting Data

The first representation (Figure 7.1a., simply consists of unstructured meeting data: the final meeting document and the recording of the audio signal, which can be accessed through standard random access functionalities. As we have seen, information in audio format is very difficult to access (Nakatani et al. 1998), for a lack of clear references points. There is clearly a need to provide users with a viable alternative to such a primitive meeting representation.

Using Meeting Speech Transcripts

A common alternative is the one depicted in Figure 7.1b.: in addition to the meeting outcome $\mathcal{O}$, textual transcripts of participants' conversations are also provided. Because of the prohibitive cost of generating transcripts manually, these are likely to be produced through the use of ASR technology. As transcripts can be time-aligned with the meeting audio recording, accessing specific parts of the original speech recording should be possible through selecting appropriate segments of the transcripts. However, speech recognition does not come without a certain cost. Although, the process is indeed automatic, transcripts remain very expensive to produce, both in terms of time (generating transcripts is often performed in dozen multiple of real-time recordings, the higher the accuracy, the highest the cost) and/or computer processing power. Even then, acceptable results can not be guaranteed, as inappropriate language models, poor recording conditions, individual speakers' accents can cause dramatic reductions in the recognition rates (Furui 1999). Even in scenarios in which ASR transcripts are of high quality, the problem still remains of finding relevant information among a high volume of information. This is clearly illustrated by Appendix B, which displays ASR transcripts of a one-hour remote meeting between two participants, which add up to 23 A4 pages! Text-based information retrieval operations traditionally used in the fields of text retrieval, topic detection and automatic summarisation can be applied to the transcripts. However, spontaneous speech is entirely different from written text and will typically contains many lexical and grammatical mistakes, false starts, repetitions, redundancy, etc. As a result, transcripts will generally require considerable processing before text retrieval operations can be performed (see chapter 2 for an overview of these techniques). Once again, these operations can be costly and the quality of the final results can not be guaranteed, as they will generally be dependant on the quality of the ASR transcripts in the first place.

Using Meeting Interactions

The third representation can be seen in Figure 7.1c.: in addition to the outcome $\mathcal{O}$, the meeting data consist of two streams of time intervals, (i) speech intervals and (ii) action intervals. Little is known about the speech intervals stream, because this knowledge is either too costly to acquire or too unreliable, or both. On the other hand, the RECOLED capture environment described in Chapter 5 provides us with comprehensive information about action intervals: their *nature*, the

agent, the *time* and *duration*, their *content* and finally, their *context* (target). Thus, our approach in the remainder of this thesis will be to *infer* information about speech intervals using information we know is *contained* in action intervals, whenever this is feasible, in order to gain access to specific points of the meeting recordings. The timing and semantics of participants' interactions are thus used to index and access the audio recording. This technique can be described as content-based access "by proxy": speech content is inferred through causal and temporal associations with well known actions. The underlying assumption is that the content and semantics of actions are mirrored in concurrent speech exchanges. This will be used as the basis for the implementation of an the interaction-based meeting browsing tool described in this chapter. Unlike speech exchanges, meeting interactions are sparse, so the trade-off of this approach consists of reliably accessing a *limited* number of *significant* meeting events. These interaction-based access techniques can also be complemented by ASR word-spotting techniques in order to improve the extent of meeting information access.

7.2.2 Semantic Persistence & Semantic Overlap

As described in the previous section, our ambition is to design a meeting browsing system which uses information contained in action intervals to infer information about speech intervals. In order to do this, we devise a simple semantic inference mechanism based on the intuitive notion of *concurrency*. Thus, the underlying idea is that the content of action intervals is likely to be reflected in the topic of *concurrent* speech intervals. However, a strict interpretation of concurrency is too constraining in practice. In remote meetings, participants may discuss a specific point for a certain amount of time. Once they have reached a decision, there will typically be a few seconds delay between the moment they stop talking (end of speech interval) and the moment they start writing (beginning of action). In this case, no semantic inference between the content of the two intervals would be possible if the inference was based on a notion of strict concurrency. This case is represented by Figure 7.2 a.), in which M(t) stands for the Membership function: if M(t) = 1 then point t belongs to the interval and if M(t) = 0, point t does not belong to the corresponding interval.

On the other hand, one could consider that speech and action intervals exhibit a certain degree of *semantic persistence, anticipation and overlap*, even though they may not be, strictly speaking, concurrent. In practice, these concepts simply reflect the notion that in meetings, participants can still be thinking of what they have just said for a small amount of time *after* they stopped talking (semantic persistence) and similarly may already be thinking of what they are about to type *before* they start typing (semantic anticipation), as there will usually be a small delay between participants reaching a decision and subsequently typing the implications down. Also, in remote meetings, participants tend not to start speaking immediately one after another in order to avoid cross-talk, and usually wait a second or two in order to make sure that the other person has

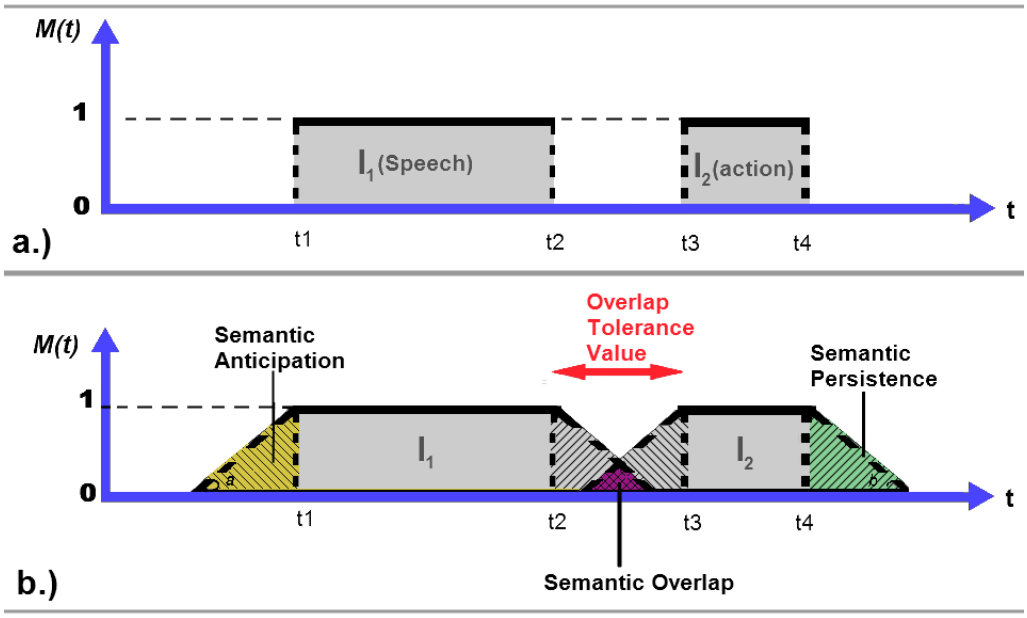


Figure 7.2: Representation of Semantic Anticipation, Persistence & Overlap using fuzzy temporal intervals

finished making his point (much like in a phone conversation). Therefore, in accordance with the natural structure of discourse, it is often reasonable to assume that media intervals in close time proximity are likely to be semantically related. Thus, semantic persistence is typically illustrated by instances of adjacent speech exchanges such as question-answer pairs, which, as highlighted by Waibel et al. (2001), are more informative jointly than in isolation. The concepts of semantic persistence, anticipation and overlap are best described using the fuzzy temporal intervals depicted in Figure 7.2 b.) in which semantic boundaries associated with speech or action intervals are gradual rather than abrupt. When intervals are "close enough", one can envisage that the combined effects of semantic persistence and semantic anticipation results in *semantic overlap* and thus, that the media intervals are likely to contain related information.

For modelling purposes, we will use the simplifying assumption that there is indeed semantic overlap between two media intervals when their intersection is above a certain (fixed) size. We make another simplifying assumption: that semantic anticipation and persistence are constant across media intervals, then the notion of semantic overlap, using fuzzy temporal intervals, can also be expressed using standard Allen (Allen 1983) intervals, but with a modified (fuzzy) *overlap* relation. We will refer to this modified *overlap* relation as "extended overlap": and as we are interested in detecting all instances of interval overlap without distinction, this new definition comprises the following Allen's relations: *during*, *starts*, *finishes*, *equals* while it extends the definition of *overlaps*, and thus *meets*.

Definition 7.1 Extended Overlap

Let I_1 and I_2 be two temporal intervals, with s_1, e_1, s_2 and e_2 respectively the start and end time of I_1 and I_2, and τ is a time threshold called the Overlap Tolerance Value (OTV). A relation of extended overlap between intervals I_1 and I_2 is defined by one of the following invertible conditions:

$$(s_1 \leq s_2) \wedge (e_2 \leq e_1)$$ *(Allen's* during *, starts, finishes and* equal *relations)*

$$OR\ (s_1 \leq s_2) \wedge (s_2 \leq e_1 + \tau)$$ *(extended overlaps, meets)*

For readability purposes and otherwise stated, we will from now on use the word *overlap* throughout the rest of this thesis when referring to the notion of "extended overlap" defined above. Figure 7.3 illustrates the concept of strict v.s. extended overlap. In a strict interpretation of temporal overlap, only the intervals I_1 and I_2 would be considered to be overlapping. On the other hand, with extended concurrency, two subsequent media intervals separated by less than the overlap tolerance value τ are also considered to be overlapping (I_2 and I_3).

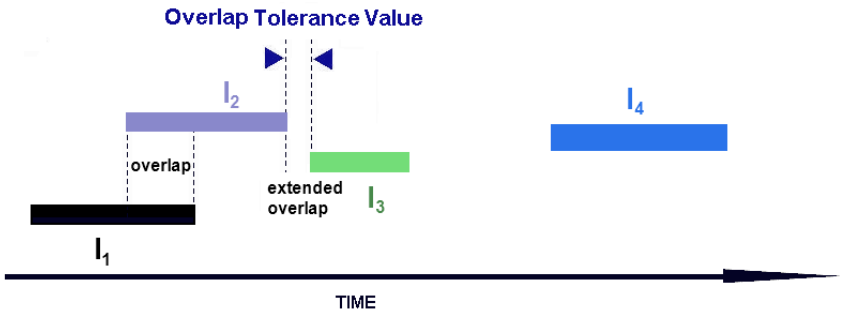


Figure 7.3: Strict v.s. Extended Temporal Interval Overlap Relationships

In practice, semantic overlap will often vary across meetings, according to individual participants and also during a specific meeting itself. We dealt with the variable nature of semantic overlap by giving users of the meeting browsing system the freedom to dynamically set the OTV value. There are also a number of occasions when topics of conversation will change abruptly. As a result, speech turns and text actions in close time proximity may occasionally be unrelated. However, the number of topic shifts during a typical meeting is small in comparison to the overall number of adjacent speech or actions intervals. Furthermore, in a large number of cases, topic shifts are indeed gradual in nature. As a result, even though the model of semantic persistence (and by association, topic shift) presented here makes a number of simplifying assumptions, it works well in practice in a majority of cases, as will be shown by the evaluation results presented in Chapter 8 and 9 of this thesis.

7.3 Content-based ("by proxy") Browsing Paradigms

7.3.1 Limitations and Strength of Interaction-based Information Retrieval

Our ambition to use information about actions as a new means to access meeting information introduces a number of limitations to the retrieval system. The first limitation is time-based and is due to the sparsity of interactions, which means that providing access to the meeting recording through information on interactions is only possible in locations when an action actually did take place. This can be interpreted both as a weakness and as a strength: a weakness because access to the recording can not be complete (lower Recall) but also a strength as meeting actions will take place during *significant* meeting events: a decision has been reached, a problem identified, important information needs to be obtained or actions taken, etc. Thus, the assumption is that accessing a meeting recording through it's interactions will provide access to a *limited* number of *important* highlights.

The second limitation is content-based: if one is to provide access to the recording through information *contained* in the interactions, then a search is only possible within the limited content of these actions, which is only an information subset of all meeting information. While this does not necessarily limit the *amount* of meeting information which can be accessed, it certainly limits the *means* of accessing it. Once again, this can be perceived as both a weakness and a strength, a weakness because of the obvious limitations of the search possibilities and a strength because one can assume that the content of actions are likely to comprise a condensed version of the most important elements of the meeting.

7.3.2 Controlled Vocabulary

We shall start with defining our use of the terms "Controlled Vocabulary", as the concept has a variety of meanings depending on it's domain of application. We believed the concept was originally crafted in the field of Information Management. Specifically, a "Controlled Vocabulary" was a finite set of definitions librarians could use to index books by providing a brief description of a book content. In Medical Informatics, the concept essentially refers to a medical terminology (Cimino 1998, Rector 1999), whose aim is to enable interoperability between medical records and systems across countries, different hospitals in the same country, or even separate wards of a given hospital. van Rijsbergen (1979), in his book "Information Retrieval", describes the notion in the specific context of some constraints imposed on indexing:

"...the vocabulary of an index language may be controlled or uncontrolled.
The former refers to a list of approved index terms that an indexer may use...".

In this thesis, we will use the latter notion of a Controlled Vocabulary as constraints imposed on the indexing terms used by the meeting browsing system. However, the controlled vocabulary is dynamic in the sense that the the constraints are only imposed to the meeting browsing system *a posteriori* and not during the meetings themselves and it does not in any way refer to a static list of terms which participants have to used during the meetings themselves. In other words, participants are free to use any terms they like during a meeting. However, the meeting browsing system will only use as index terms the words found in the meeting document outcome $\mathcal{O}$. The assumption here is that meeting actions will generally contain some of the most important elements of a meeting. Thus, the most important meeting *keywords* are also likely to be found in the final meeting outcome. Thus, we define an indexing Controlled Vocabulary $\mathcal{V}$ as the set of all words in the meeting outcome (either loaded at the beginning of the meeting or typed at some stage during the meeting), less a list of stop-words.

Definition 7.2 Controlled Vocabulary

Given a meeting document outcome $\mathcal{O}$ and a list of stop words $\mathcal{L}$

The Controlled Vocabulary $\mathcal{V}$ *is defined as the set of words ω:*

$$\mathcal{V} = \{\ \omega \mid (\omega \in \mathcal{O}) \wedge (\omega \notin \mathcal{L})\}$$

There are many reasons why such a definition of a Controlled Vocabulary is appealing:
(i) it is reasonable to assume it will contain all the most important words used during the meeting (ii) it resolves a number of issues relating to important meeting terms, such as named entities (people and places), which could otherwise easily be left as Out-Of-Vocabulary (OOV) items in ASR transcripts: named entities, as long as they are present in the final outcome will effectively be indexed, (iii) a by-product of the previous rule is that if a participant regularly uses abbreviations (i.e. *Sat night* for Saturday night), or consistently misspells the same word, these will also be indexed (iv) as the number of words in $\mathcal{V}$ is limited, keyword matching will be less computationally expensive, allowing for efficient search.

7.3.3 Keywords Search

Once punctuation, figures and stop words have been removed from $\mathcal{V}$, a set of potential keywords has been identified by the system. We can then implement an interaction-based Keyword Search function $\mathcal{KS}()$ by using string-matching on the content of meeting actions and returning the set of all audio segments in the vicinity of editing actions which were found to contain a specified keyword. In the following definition of a Keyword Search function, S_j represents a speech interval and Terms(S_j) the set of terms uttered during this speech interval.

Definition 7.3 Keyword Search

Given a set $\mathcal{T} = \{T_1, ..., T_{|\mathcal{T}|}\}$ *of text actions (temporal intervals), let Terms*(T_i) *be the set of terms contained in the text action* T_i.

Given a set $\mathcal{S} = \{s_1, ..., s_{|S|}\}$ *of speech segments (temporal intervals), let Terms*(S_j) *be the set of terms uttered during the speech interval* S_j.

Given a specific term $\mathcal{K}w$*, a Keyword Search is a function* $\mathcal{KS} : \mathcal{V} \rightarrow \mathcal{S}$ *which returns a set of speech segments:*

$$\mathcal{KS}(\mathcal{K}w) = \{s_i \in \mathcal{S} \mid (s_i \textbf{ overlaps } T_j) \ \wedge \ (\mathcal{K}w \in Terms(T_j))\}$$

Note that this search method does not rely on speech recognition but only on the content of text actions. As a result, it is not subject to traditional ASR shortcomings such as disfluencies in spontaneous dialogues, poor recording conditions, crosstalk, inappropriate language models, out-of-vocabulary items and variations in speaking styles and pronunciations which means that for a certain percentage of people, some speech recognition systems may have very low recognition rates (Furui 1999). There is however an underlying assumption behind this search method: which we have already referred to as the "content-based by proxy" assumption:

Assumption 7.1 *"When participants manipulate (e.g: type, delete or point at) a specific term, they are likely to have uttered this word (or otherwise a synonym) shortly beforehand or are about to utter it shortly afterwards."*

Note that this assumption is only made from Text-to-Speech: if a word is typed or pointed at, then it is likely to be also uttered. We do not make the reverse and counter-intuitive assumption which would consist in stating that when a word is spoken, it is also typed in close temporal proximity. The Text-to-Speech assumption is illustrated in Figure 7.4 and we will see in the analytic evaluation presented in Chapter 8 the extent to which this assumption is valid in practice.

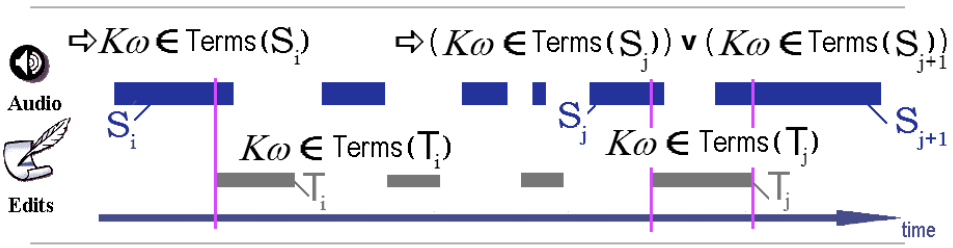


Figure 7.4: The concurrency of typed and spoken keywords assumption: the assumption (represented by arrow) is only made from Text-to-Speech

Stemming

In order to improve the efficiency of a keyword search, we mapped all keywords in $\mathcal{V}$ to their common stems, whenever necessary. We used the suffix stripping algorithm first developed by Porter (1980), in order to find words with related stems and implement it as follows:

Algorithm 7.1

Mapping Related Keywords through common stems

→ apply Porter's algorithm to all terms $\mathcal{K}w_i \; of \; \mathcal{V} : Porter(Kw_i)$

→ apply Porter's algorithm to the remaining terms $Kw_j \; (i \neq j) \; of \; \mathcal{V} : Porter(Kw_j)$

→ map related keywords to same stem whenever a match is found
(i.e. $Porter(Kw_i) == Porter(Kw_j)$)

Although this stemming algorithm only applied to a minority of words due to the limited size of the Controlled Vocabulary $\mathcal{V}$, it is critical in providing adequate search functionality in certain cases. In particular, it ensures that a specific keyword is identified in the meeting recording regardless of it's number (e.g. *day* and *days*, *ticket* and *tickets*) or form (e.g. *play*, *played* and *playing*).

7.3.4 Complex Query - Topic Search

The main limitations of a Keyword Search-based query is that a significant number of the terms in the Controlled Vocabulary are semantically poor on their own. While named-entities such as a place (*Dublin*) or person (*Seamus*) describe very specific information, and therefore may be sufficient to identify some information of interest on their own, a majority (e.g. "one", "day", "night", "euro", etc.) are too general to be useful on their own in a query. On the other hand, a combination of these words may be able to narrow down the possible candidates to a handful of locations within the meetings. As an example, the combination of "day" and "one" may be able to distinguish the *first day* of the trip from: (i) all the other days and (ii) all the single items mentioned in a meeting recording.

Thus, we simply define a complex query as a conjunction of keywords searches: and further distinguish between two special cases of search: OR-Search and AND-Search. We will subsequently refer to the complex query as the Topic Search functionality [1].

[1] Our assumption is that co-location of terms in participants' editing actions potentially suggest a shared meaning between these terms and that the semantics will be reflected in participants' concurrent speech turns. However, we accept that even though co-located terms may indeed be semantically related, this will not always translate in these terms being the actual "topic" of the conversation

Definition 7.4 OR - Search

A Topic "OR" Search is a function $\mathcal{TS}_{or} : \mathcal{V} \rightarrow \mathcal{S}$ *which returns for the keywords* $(Kw_1, ..., Kw_j)$ *a set of speech segments such as:*

$$\mathcal{TS}_{or}\ (\mathcal{K}w_1 \vee ... \vee \mathcal{K}w_j) = \{\ KS\ (\mathcal{K}w_1) \cup ... \cup KS\ (\mathcal{K}w_j)\ \}$$

Definition 7.5 AND - Search

A Topic "AND" Search is a function $\mathcal{TS}_{and} : \mathcal{V} \rightarrow \mathcal{S}$ *which returns for the keywords* $(Kw_1, ..., Kw_j)$ *a set of speech segments such as:*

$$\mathcal{TS}_{and}\ (\mathcal{K}w_1 \wedge ... \wedge \mathcal{K}w_j) = \{\ KS\ (\mathcal{K}w_1) \cap ... \cap KS\ (\mathcal{K}w_j)\ \}$$

The definition of the OR - Search is reasonably straightforward, it is a conjunction of keyword searches, and returns speech segments in close time proximity of text actions containing any of a number of keywords. The AND - Search is a more delicate issue: in this case, we are interested in a number of keywords which need to be present in the same, or at least very close, speech segments. Thus, we once again use the notion of semantic overlap, described earlier in this chapter, to infer semantic relatedness between neighbouring media intervals which contain the terms of the query within a certain acceptable time threshold (Topic Span Threshold), beyond which semantic relatedness is deemed unlikely. Thus, the AND - Search relies on the following assumption:

Assumption 7.2 *"In the Topic Search* $\mathcal{TS}_{and}\ (\mathcal{K}w_1 \wedge ... \wedge \mathcal{K}w_j)$*, the keywords* $Kw_1, ..., Kw_j$ *are more likely to be semantically related if they were typed in close time proximity, and the shorter the delay, the more likely the existence of a shared meaning".*

Note that Assumption 7.2 does not state that time difference can be used as a measure of semantic relatedness for *any* pair of keywords in $\mathcal{V}$ (e.g. "kettle" and "fish"). What it says instead is that if the keywords $Kw_1, ..., Kw_j$ have been selected as part of a user query, then the time difference may be used to check whether occurrences of these words are close enough to justify a possible link, thus potentially providing the information the user is looking for. In other words, the fact that the user has chosen to type these words in a single query is interpreted in itself as a semantic *likelihood* given a certain context (e.g. prior knowledge about the meeting, some information expectation or simply common sense). The time difference between the typed occurrences of these terms is then subsequently used as a measure of this semantic likelihood, with a short time indicating a strong likelihood and a long time a low likelihood. This is illustrated in Figure 7.5, where τ represents the Topic Span time Threshold, beyond which semantic likelihood is considered unlikely, and *sem*(Kw_a, Kw_b) is a boolean function which returns true if Kw_a and Kw_b are part of the same topic and false if they are not.

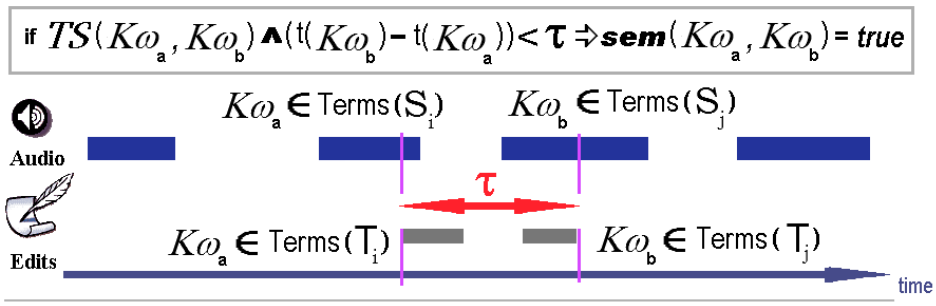


Figure 7.5: The Topic Assumption: time distance between occurences of words in a Topic Search $\mathcal{TS}_{and}\ (\mathcal{K}w_a, \mathcal{K}w_b)$ is used as a measure of semantic likelihood between these keywords.

7.3.5 Action-based browsing

Finally, one can use the content of participants' actions as a browsing modality. Space-based interactions can be interpreted as yet another (discrete) media stream, thus sharing with the audio stream the time dimension. This can be exploited in two ways for meeting browsing. The first one consists of displaying the content of concurrent actions while playing the time-based media. In figure 7.6, listening to audio segment Au_1 would prompt the text "*180 per person*" to appear. In other words, it consists of the synchronous play-back and display of multiple media streams, as previously implemented in a number of multimodal browsers (Geyer et al. 2003, Brotherton et al. 1998, Wellner et al. 2004). However, the time relationship between the media can also be exploited by visualising the space-based interactions as a navigation tool into the time-based media recording. Using a slide bar to navigate the timeline will display concurrent space-based actions, thus offering hints about the content of surrounding audio segments. In Figure 7.6, the navigator will prompt the display of the message "*180 per person*", which suggest that surrounding speech exchanges deal with costs (*of a room*). The definite appeal of such a navigation method is that space-based actions will generally have strong associated semantics and are appropriate for quick visual scanning, thus potentially offering a powerful indexing method into the time-based media. Interactions are discrete, generally sparse enough so as not to overload a user with information, and tend to form natural semantic clusters over time (when a specific topic is discussed) allowing for discrimination and segmentation of topics within a meeting recording.

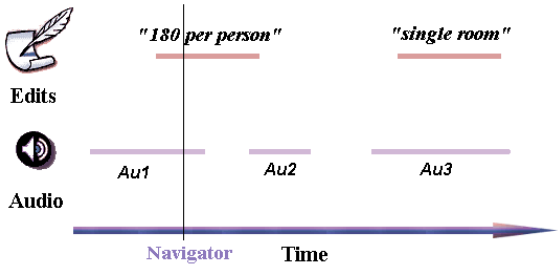


Figure 7.6: Using the content of Edits as an audio navigation tool

7.3.6 Limitations of Content-Based Search Functionalities

The keyword and topic search described so far in this chapter are essentially based on "search by content": in which the content (terms) of an action or speech turn is used in order to identify the potential locations of some information of interest in a meeting recording. The content-based search techniques provide access to the time-based media on a punctual basis: by locating regions in the speech recording where a word or combination of words are likely to have been uttered. Using the archer metaphor presented in chapter 4, this is akin to identifying sections of the meeting by looking at the time at which specific arrows were shot. While this type of search will generally work reasonably well in practice, none-the-less because the search concept is both intuitive and familiar to users, there are a number of cases when content-based search can be potentially misleading or simply fail to identify relevant information. Consider this example from one of the meetings of the evaluation subset (described in more detail in section 8.2.2), in which participants were asked to organise an itinerary of an eleven day tour of China. As a way of structuring the meeting, participants first entered early in the meeting a list of items: "Day 1", "Day 2", ... up to "Day 11". The activities of each of these days were then subsequently discussed one by one as the itinerary was slowly put together. In this case, searching for a specific day such as "Day 7" using the text-based content search would not point to speech turns relating to the activities of the seventh day but simply to the time when the list of days was first typed.

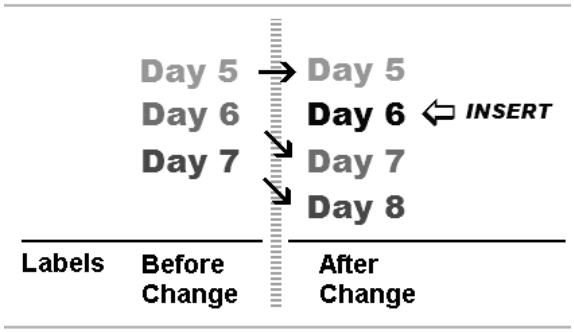


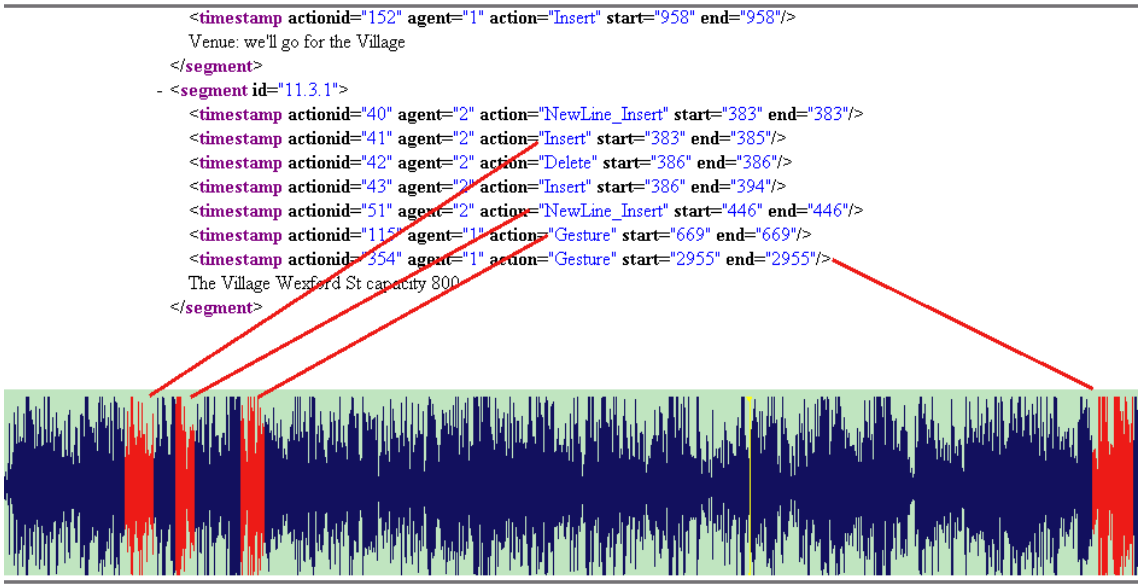
Figure 7.7: A numbering change in the course of the meeting can result in potentially misleading Keyword searches

Furthermore, numbering of the days were also changed during the course of the meeting as the participants made some modifications to the itinerary, adding an extra day in the middle of the trip, as illustrated in Figure 7.7. Thus, in a scenario where ASR transcripts were flawless, searching for information "Day 7" using a speech-based content search would return speech turns containing the words "Day 7". This search result would thus consists in a combination of speech turns relating to what *was* labelled Day 7 before the change (and is now Day 8) and what is now Day 7 (and was labelled Day 6). This set of information is not a satisfactory search outcome, as the information provided relates to two *distinct* days with the same label and could potentially

lead to some confusion. This example serves to illustrate how content-based search will in special circumstances miss relevant information while retrieving potentially misleading information.

7.4 Context based Browsing

A possible alternative to a content-based search is one we will refer to as a *contextual* search (paragraph-based). In the contextual search, instead of looking at the content of participants' actions in isolation (the arrows), one looks at the context (target) of these actions (interactions with meeting document). In this perspective, potential links between different type of actions, performed at arbitrary times and possibly by several agents, can be highlighted based on their shared context (e.g. a paragraph of the document). We argue that this type of information linking can potentially uncover semantic relations between separate pieces of meeting information, which could not necessarily be inferred by using content-based or time-based techniques alone.



-
<timestamp actionid="40" agent="2" action="NewLine_Insert" start="383" end="383"/>
<timestamp actionid="41" agent="2" action="Insert" start="383" end="385"/>
<timestamp actionid="42" agent="2" action="Delete" start="386" end="386"/>
<timestamp actionid="43" agent="2" action="Insert" start="386" end="394"/>
<timestamp actionid="51" agent="2" action="NewLine_Insert" start="446" end="446"/>
<timestamp actionid="115" agent="1" action="Gesture" start="669" end="669"/>
<timestamp actionid="354" agent="1" action="Gesture" start="2955" end="2955"/>
The Village Wexford St capacity 800

Figure 7.8: Using contextual information for accessing a meeting recording

7.4.1 Simple Paragraph-based Retrieval

In the paragraph-based retrieval approach, the meeting document $\mathcal{O}$ (outcome) is used as the underlying structure for contextual retrieval, while the individual elements of this information structure are the paragraphs of text. Using the meeting outcome and it's paragraphs seemed to be a natural and intuitive starting point to explore the recording. The assumption here is that

meeting participants will, to some extent, naturally use text segmentation (chapters, paragraphs, sentences etc...) to structure their ideas in semantic units. Thus, as paragraphs were selected as the fundamental units for interaction logging, a simple contextual search initially maps all actions performed on a single text entity to all corresponding speech turns. This simple information linking is illustrated in Figure 7.8.


-
<timestamp actionid="28" agent="1" action="NewLine_Insert" start="120" end="120"/>
<timestamp actionid="29" agent="1" action="NewLine_Insert" start="121" end="121"/>
<timestamp actionid="102" agent="1" action="Insert" start="662" end="663"/>
<timestamp actionid="103" agent="1" action="Delete" start="664" end="664"/>
<timestamp actionid="104" agent="1" action="Insert" start="664" end="665"/>
<timestamp actionid="265" agent="2" action="Paste" start="2358" end="2358"/>
<timestamp actionid="266" agent="2" action="Insert" start="2362" end="2362"/>
<timestamp actionid="267" agent="2" action="Insert" start="2369" end="2369"/>
<timestamp actionid="268" agent="2" action="Cut" start="2375" end="2375"/>
<timestamp actionid="280" agent="2" action="Paste" start="2473" end="2473"/>
<timestamp actionid="281" agent="2" action="Insert" start="2478" end="2478"/>
<timestamp actionid="304" agent="2" action="Insert" start="2682" end="2683"/>
<timestamp actionid="305" agent="2" action="Delete" start="2683" end="2684"/>
<timestamp actionid="306" agent="2" action="Insert" start="2685" end="2689"/>
<timestamp actionid="367" agent="1" action="Delete" start="3389" end="3389"/>
<timestamp actionid="368" agent="1" action="Insert" start="3389" end="3389"/>
<timestamp ="373" agent="1" action="Delete" start="3409" end="3409"/>
<timestamp ="374" agent="1" action="Insert" start="3409" end="3409"/>
DAY7: fly to shanghai early morning flight (13million people) interesting architeture (the Bund)

<action id="365" type="Delete" startT="3386" endT="3386" paragraphs="25" startOffset="6" endOffset="7"> 5 </action>
<action id="366" type="Insert" startT="3386" endT="3386" paragraphs="25" startOffset="6"> 6 </action>
<action id="367" type="Delete" startT="3389" endT="3389" paragraphs="26" startOffset="3" endOffset="4"> 6 </action>
<action id="368" type="Insert" startT="3389" endT="3389" paragraphs="26" startOffset="3"> 6 </action>
<action id="369" type="Delete" startT="3392" endT="3392" paragraphs="27" startOffset="6" endOffset="7"> 6 </action>
<action id="370" type="Insert" startT="3392" endT="3392" paragraphs="27" startOffset="6"> 6 </action>
<action id="371" type="Delete" startT="3394" endT="3394" paragraphs="28" startOffset="3" endOffset="4"> 7 </action>
<action id="372" type="Insert" startT="3396" endT="3396" paragraphs="28" startOffset="3"> 7 </action>
id="373" type="Delete" startT="3409" endT="3409" paragraphs="26" startOffset="3" endOff 6 </action>
id="374" type="Insert" startT="3409" endT="3409" paragraphs="26" startOff 7 </action>
<action id="375" type="Delete" startT="3412" endT="3412" paragraphs="27" startOffset="5" endOffset="6"> 6 </action>
<action id="376" type="Insert" startT="3413" endT="3413" paragraphs="27" startOffset="5"> 7 </action>
<action id="377" type="Delete" startT="3417" endT="3417" paragraphs="28,30" startOffset="6" endOffset="0"> Fly to Yunnan </action>
<action id="378" type="Paste" startT="3418" endT="3418" paragraphs="31,31.1" startOffset="4"> Fly to Yunnan </action>
<action id="379" type="Cut" startT="3420" endT="3420" paragraphs="28" startOffset="0" endOffset="6"> DAY7 : </action>

Figure 7.9: Paragraph history permits to track down contextual modifications such as labelling changes

The implications of such simple contextual information linking are best illustrated by revisiting the example used in the previous section 7.4.3 to describe the limitations of content-based search. As interactions are recorded at the paragraph level, all modifications on a specific text segment have been recorded. Thus, it is possible to retrieve all actions relating to a specific paragraph, including cases of label modifications: when participants were discussing of the same semantic entity (i.e. the day in Shanghai), albeit with a different label (when it was still day 6). Figure 7.9 illustrates how the labelling change has been recorded in the paragraph log (upper half of the Figure) and with the corresponding actions displayed in the bottom half. Thus, a contextual search performed on this paragraph would return a combination of speech turns relating to what *was then* labelled Day 6 before the change (and is now Day 7) and Day 7. This is the correct outcome as all information provided is related to the *same specific* day (in Shanghai). As a result, accessing information from the context's (paragraph) perspective is both consistent with participants' intentions and also reflects the natural evolution of ideas during meetings. In this example, it is interesting to note

that as the re-labelling was the last action performed on this specific paragraph, a content-based search would have missed the vast majority of relevant information.

7.4.2 Extended Paragraph-based Retrieval: Temporal Neighbourhood

While one can retrieve all speech turns concurrent with modifications performed on a specific paragraph, one could argue that concurrent interactions which were performed on other parts of the document around the same time can potentially provide additional contextual information about participants' actions and decisions. Thus, the simple information linking described in the previous section can be further expanded to include additional contextual information by using the notion of Temporal Neighbourhood described in chapter 4 and Semantic Overlap described in this chapter. As a result, we propose the following algorithm for paragraph-based retrieval:

Algorithm 7.2

Temporal neighbourhood paragraph retrieval.

→ retrieve the set of all action timestamps a_i performed on a specific paragraph P_n

→ retrieve the set of all speech segments s_j overlaping with the actions a_i

→ retrieve all action timestamps a_k performed on *different* paragraphs P_m $(m \neq n)$ which took place within the duration of the previous set of speech segments s_j

→ iterate through the 2 previous steps until no new actions or speech intervals can be found

---

Figure 7.10 illustrates how the temporal neighbourhood paragraph retrieval algorithm operates in practice: in the first step (bottom editing operation layer), all the actions which affected a specific paragraph are retrieved. In the following step, all concurrent speech exchanges are included (one level up: speech layer 1). Then, actions performed on *other* paragraphs which occurred during the previous speech exchanges are retrieved (editing layer 2). These can possibly lead to further speech exchanges (speech layer 2) which will potentially point to further actions (editing layer 3).

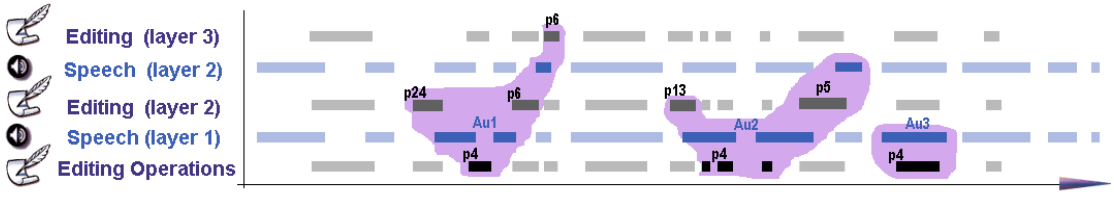


Figure 7.10: Paragraph Temporal Neighbourhood Retrieval

A paragraph's temporal neighbourhood is therefore a tree-shaped information structure: with the paragraph as the root of the tree, actions performed on this paragraph as it's direct children,

the concurrent speech turns as children of the action nodes and actions performed on *other* paragraphs during these speech turns as further children nodes of these speech turns, etc. Figure 7.11 is an example of the visual representation of a paragraph's temporal neighbourhood. An important parameter in shaping the tree is to what degree one considers actions and speech turns (temporal intervals) to be concurrent (i.e. what is the Overlap Tolerance Value?) An OTV set to zero (strict concurrency) ensures that information retrieved is generally directly related to the manipulation of a specific paragraph, thus ensuring good precision but possibly missing related peripheral information. Inversely, increasing the tolerance value may increase recall but at the expense of including irrelevant information. In practice, we found that the OTV could vary widely depending on the type of meetings: a small OTV works well with fast meetings with a lot of interactions while slower meetings with fewer actions warrant a higher OTV. As a result, the OTV in the Meeting Miner is a parameter which can be adjusted by the users, and with a default value of 5 seconds.

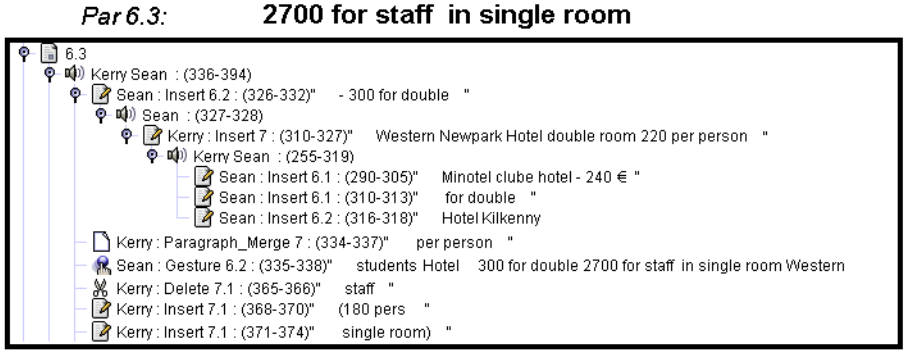


Figure 7.11: A paragraph Temporal Neighbourhood

One of the most interesting aspects of the paragraph temporal neighbourhood retrieval is that it can potentially highlight information linking between non-adjacent parts of the meeting outcome, which could have been modified at arbitrary times during the meeting, as long as they were, at some stage, subject to some (time-based) overlapping modifications (Figure 7.12).

7.4.3 Noise as Limitations of Time-based Contextual Information Representation

One of the most challenging aspects of making use of contextual information is providing useful information to the users in a readable and concise format. Figure 7.13 is a typical example of information overload: it displays a "raw" paragraph temporal neighbourhood, where the algorithm 7.2 has been applied without any type of information filtering. It is very hard to make any sense of the information displayed.

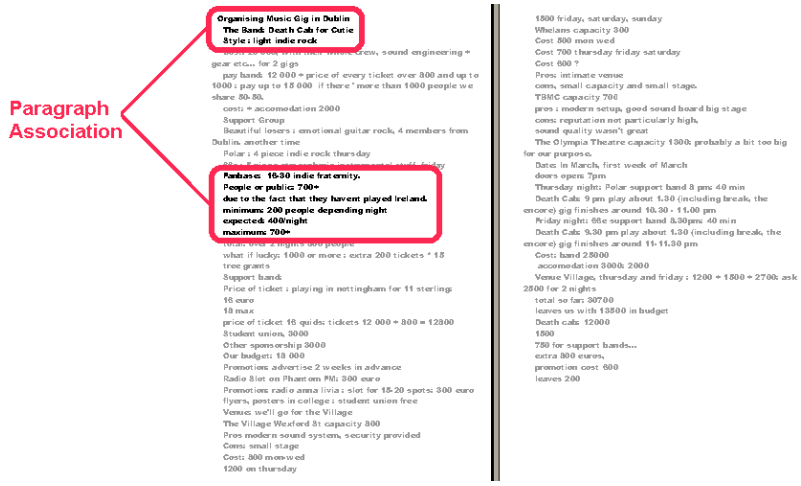


Figure 7.12: Paragraph association highlighted in the meeting outcome through a paragraph-based temporal neighbourhood retrieval

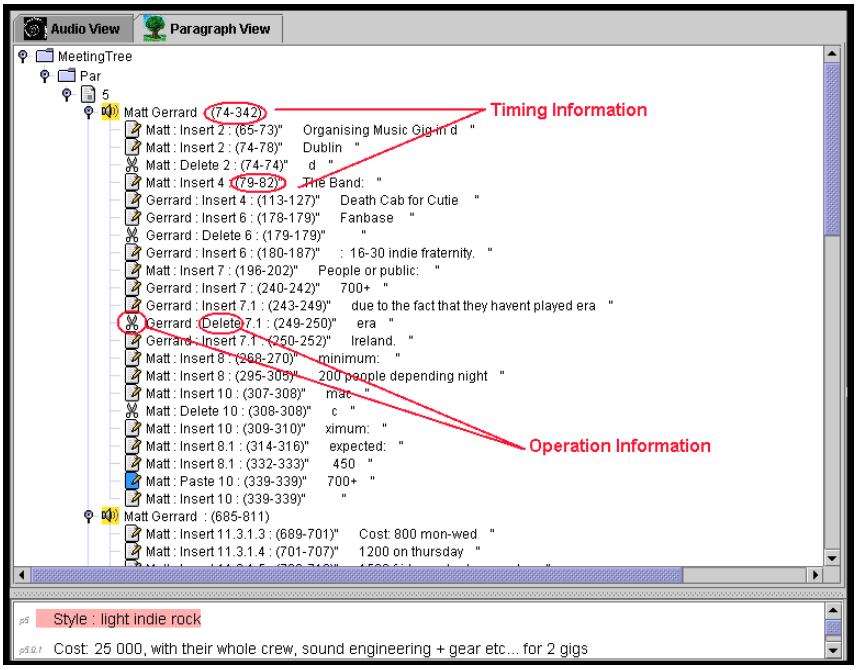


Figure 7.13: Contextual Information Overload

Steps taken to improve the display of Contextual information

There are a number of steps which can be taken in order to improve the display of contextual information.

Reduce the Number of Speech Turns Displayed There are usually a large number of speech exchanges during meetings, typically in the hundreds, even for relatively short (20 to 45 minutes) meetings (Luz and Bouamrane 2006). A significant number of these speech turns are of very short duration and almost always consist of false starts, acknowledgements or comments made while another participant is speaking (e.g. “yes”, “right”, ”o.k.”). It can be

argued that in remote collaborative meetings, participants feel a greater need to acknowledge each other's presence with verbal utterances as a way of maintaining awareness (Dourish and Bellotti 1992) in remote collaborative meetings. As a result, extra care is required when visualising such a rich set of speech turns. Figure 7.14 illustrates this phenomenon of high occurrences of acknowledgements in remote meetings: with no less than 21 instances of verbal utterances (including 16 "yeah") in less than 3 minutes.

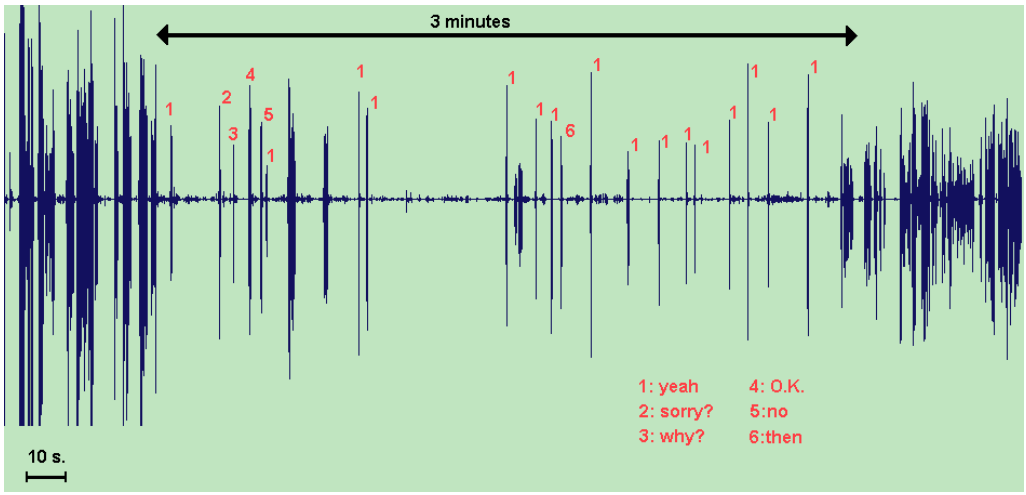


Figure 7.14: The high level of verbal acknowledgements in remote meetings

Taken out of context these individual utterances would appear to be meaningless. Therefore, in order to prevent them from being individually displayed, all audio segments related by the *during* relation are merged in a single speech node in the contextual information display. This more compact audio mapping means that if a participant makes a comment while another participant is speaking, these concurrent speech exchanges are graphically displayed on the interface as a single audio segment. In order to highlight the fact, that several participants are speaking during this audio segment, the names of all active speakers are displayed beside the corresponding audio node.

While this first step is useful in achieving a more compact audio mapping, it is somehow insufficient and further audio amalgamation is necessary for compact graphical representation. Thus, audio intervals related by the *overlaps* relation are also subsequently amalgamated into a single speech node. In accordance with the natural structure of discourse, it is reasonable to assume that audio segments in close time proximity may be relevant to one another (the semantic overlap described earlier in this chapter). This can be viewed as the question-answer pair paradigm (Waibel et al. 2001) where adjacent speech exchanges are more informative jointly than when considered on their own. The two steps taken for compact graphical representation are illustrated in Figure 7.15. In theory, if participants' speech turns constantly overlapped one another, the audio merging algorithm proposed could result in the meeting

being reduced to a single speech interval. In practice, this never happens, showing that natural pauses in discourse are indeed a simple, yet efficient, way of segmenting speech in coherent semantic units, as shown by other studies (Pfeiffer 2001).

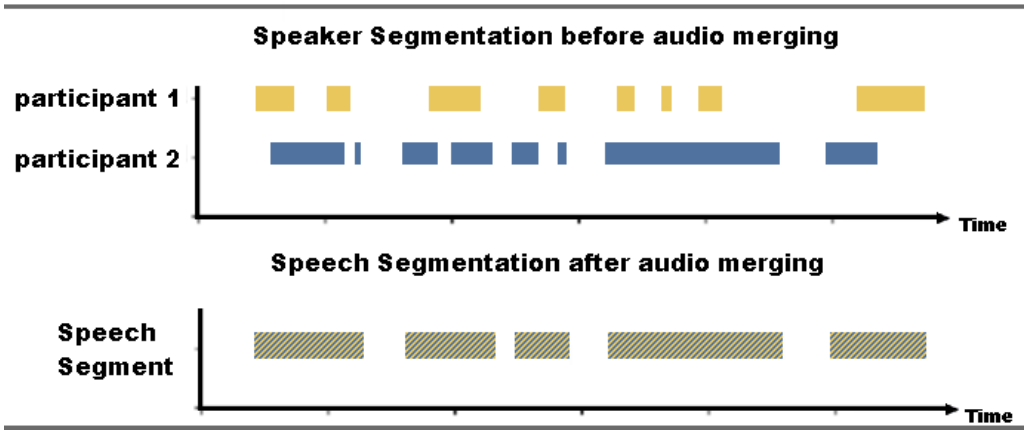


Figure 7.15: Speech intervals amalgamation for compact graphical representation of contextual information

Reduce Redundancy in information Displayed Other issues in the contextual information displayed in Figure 7.13 is the high degree of redundancy in the information. While timing information associated with speech segments is important to provide location information within the recording, the usefulness of displaying the time of individual actions is less obvious as long as one knows that the action took place within the duration of a specific speech interval. As a result, the timing information associated with actions is removed from the graphical display. The other redundancy is related to the nature of participants' actions, which is originally indicated both by an icon and in writing. As icons are sufficiently explicit to describe the type of actions (Figure 7.16), the textual descriptions of the nature of actions is subsequently removed from the graphical representation of contextual information.

ICON	DESCRIPTION
	Audio segment
	Paragraph
	Insertion
	Paste
	Deletion
	Cut
	Point

Figure 7.16: The icons used for items and operations descriptions in the user interface, (Open Source: http://wm-icons.sourceforge.net/)

Finally, when the same agent is responsible for a number of consecutive actions, which is often the case in practice as participants usually carry actions in spurts, his name is only displayed on the first action. Figure 7.17 displays a paragraph Temporal Neighbourhood once the previous steps to remove information redundancy have been performed.

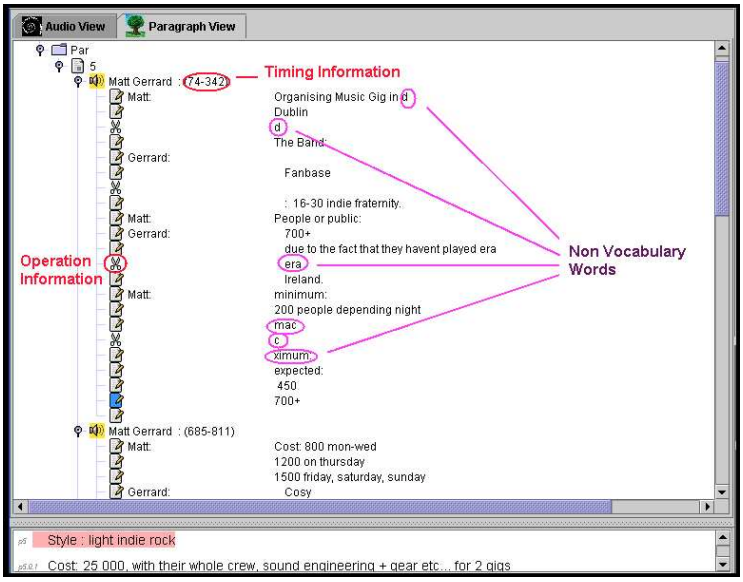


Figure 7.17: Paragraph Temporal Neighbourhood with information redundancy removed

Remove non-Vocabulary words The final step taken to improve the readability of contextual information consists of removing non-vocabulary words from the information graphical representation. When typing, participants often make typographic or spelling mistakes, which they will often immediately correct whenever they become aware of a mistake. This will often mean that individual edits may contain single letters or a few letters (deleted or replaced) and misspelled words as shown in Fig. 7.18. In order to identify meaningful words, one simply uses the Controlled Vocabulary $\mathcal{V}$ defined earlier in section 7.3.2. Thus the algorithm used to retain meaningful word is simple: if the word tokens contained in the individual actions are not present in $\mathcal{V}$, they are discarded.

Raw Edit	Extracted Keywords
Weekend in Hotel KJi Kilkenny : Vist travel bu c m, by hired bus 25 earch Sat Night	Weekend Hotel Kilkenny travel hired bus Sat Night

Figure 7.18: Raw content of edits and extracted action keywords

Figure 7.19 shows a paragraph contextual information once all the previous information filtering steps have been taken. This results in a much neater and concise display in which larger speech segments are associated with corresponding editing information, thus providing hints about the likely content of speech. The list of actions associated with the first speech node clearly indicates that the context of this speech segment is related to general information about organising a concert. The list of actions associated with the second speech node on the other hand clearly indicates that the topic of the meeting has by then switched to issues of cost and dates.

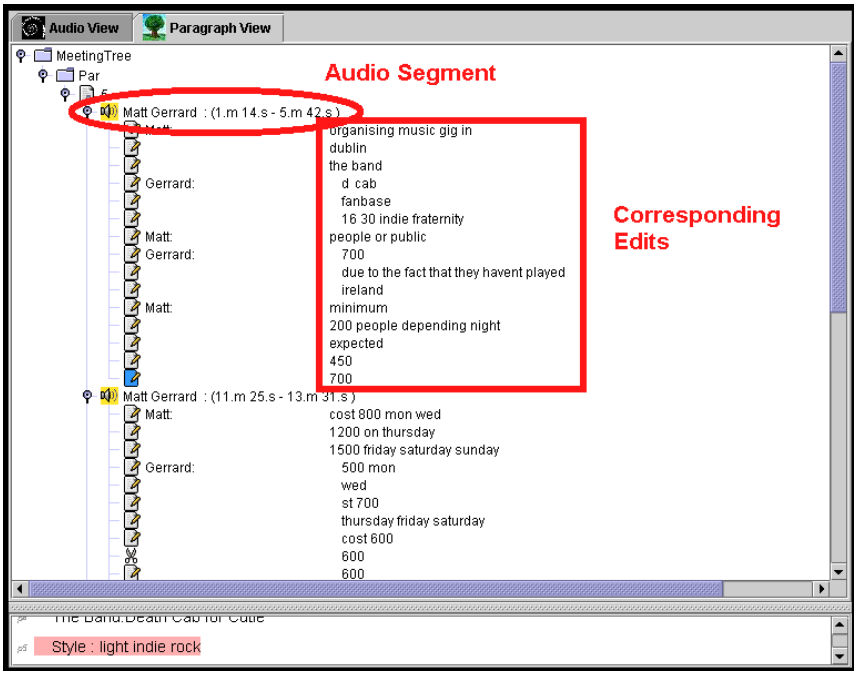


Figure 7.19: Paragraph Temporal Neighbourhood as speech nodes with associated contextual information

7.5 Meeting Miner implementation: the User Interface

As we have presented in details the underlying search, browsing and navigation mechanisms of our interaction-based meeting navigation system, the Meeting Miner, we will now describe the various functionalities and components of the browser's graphical user interface. The Meeting Miner was implemented using the Java programming language and the main components of the original user interface are described in Figure 7.20. As we will see in the user-centred evaluation in chapter 9, the heuristic evaluation and usability study of the Meeting Miner resulted in modifications to the final functionalities and user interface of the browsing tool (Figure 7.21).

Document View displays the meeting document outcome $\mathcal{O}$ in the lower pane. Each paragraph in the document view is indexed for easy cross referencing. Participants' individual contributions can be highlighted according to a colour code. When the paragraph-based view is

selected, clicking on an individual paragraph will display the paragraph's temporal neighbourhood. Specific sections of the document can be highlighted for keyword or topic based search, which are performed according to the words selected.

Tool Bar Enables the user to adjust system settings to best suit his preferences. Parameters which can be adjusted include the Overlap Tolerance Value, keyword preferences, display preferences, etc.

Upper panel The upper pane can be one of two views, depending on how the user wishes to browse the meeting. In a paragraph-based retrieval, clicking on a specific paragraph will prompt the display of the tree-structured paragraph retrieval unit (paragraph temporal neighbourhood) consisting of the content of editing nodes, and corresponding audio nodes, with the names of all the active participants within the duration of these temporal intervals.

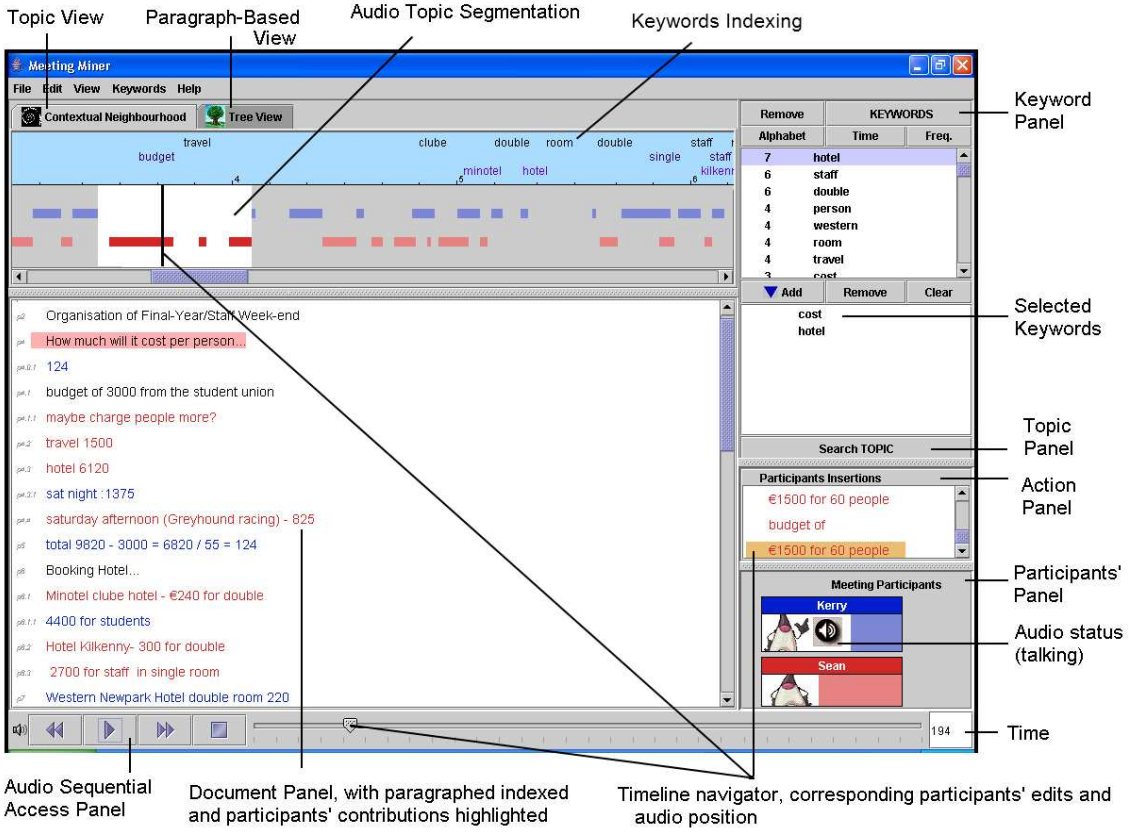


Figure 7.20: Earlier version of Meeting Miner User Interface

An alternative view is the topic view, or *contextual neighbourhood* view. When this mode is selected, regions where audio contributions are likely to be related to the topic selection (a set of keywords selected in the topic panel) are highlighted. Clicking on a particular interval will play the corresponding audio.

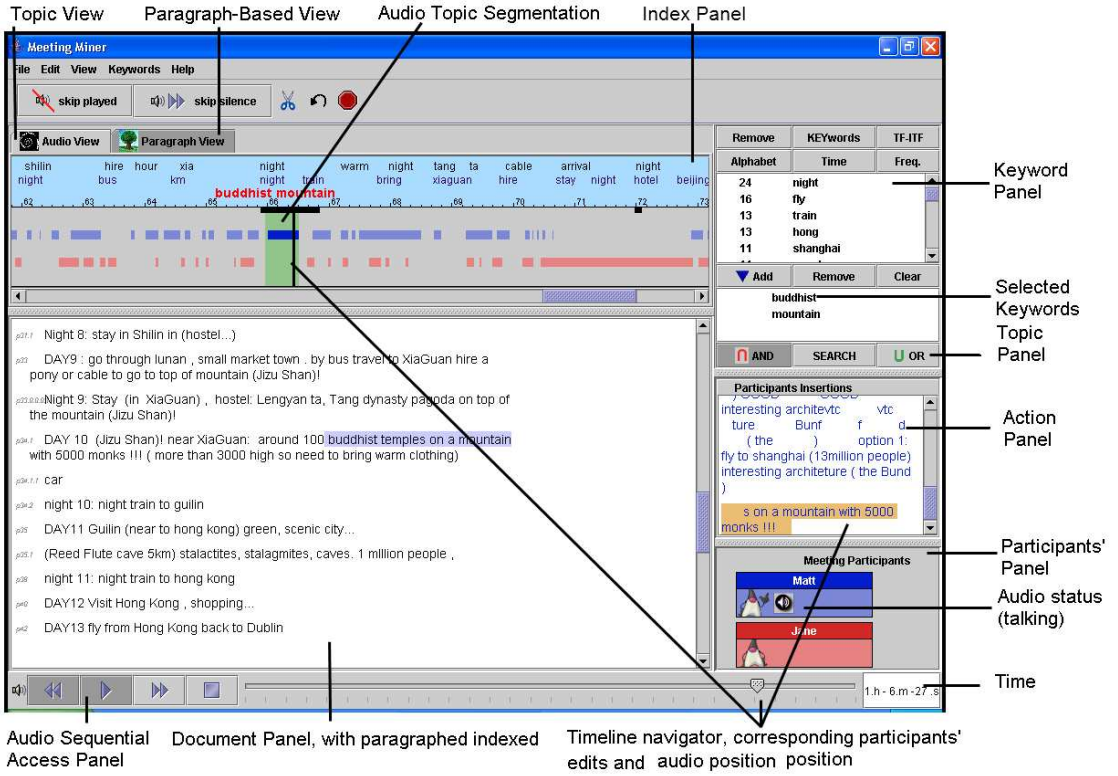


Figure 7.21: Meeting Miner User after modifications performed as the results of the usability evaluation

Keyword Panel. Displays all the potential keywords from the text document identified by the system. The list of keywords can be displayed in alphabetical order, frequency ranking or simply time of appearance. The user can dynamically update the list (removing words under a certain frequency or only select keywords associated with a certain type of action, etc.)

Topic Panel. The user can dynamically choose a set of keywords. A subsequent topic search will highlight audio segments associated with these keywords. The audio intervals selected by the topic search are segments in the neighbourhood of participants' edits which contain the keywords.

Action Panel. Used in conjunction with the timeline navigator (slider) bar, it displays the nature of concurrent participants' edits for action-based browsing.

Participants' Panel. Displays the names of the participants. Each participant is assigned a unique colour code which highlights on the interface the ownership of the various text and audio contributions. A little audio icon is also displayed to show participants current activities (speaking, idle, etc).

Audio Panel Provides sequential and random access to the audio file. The browser's audio mode settings offers the user several navigation options such as skipping silences, or, if the topic mode is selected, jumping to the next topical segments. Similar functionalities were implemented in the SpeechSkimmer (Arons 1997).

Timeline Navigator (slider) The navigator's purpose is twofold: first, it offers a reference point into the audio recording. It also offers random access to the audio file. While moving the slider, participants' concurrent actions are displayed in the Action Panel, so the user can decide to stop and listen to a specific section of the recording if he were to see an action of particular interest (as described in section 7.3.5).

7.6 Adding Speech Recognition

The main anticipated advantage of using participants' interactions as the basis for browsing meeting recordings lies in the assumption that interactions will often be associated with important meeting events: a decision has been taken, critical information needs to be collected, a course of action has been decided, etc. Thus, the potential sparsity of actions can be interpreted as a strength of the system in that they provide access to events which could in themselves be considered as a type of meetings' "highlights". Of course, it can also be argued that this is also a weakness, as parts of meeting with little or no interactions between participants (which happen if participants discuss for a significant amount of time before jolting down any decisions) become inaccessible meeting recording "black holes". However, the interaction-based navigation mechanisms presented here can naturally be used in conjunction with other complementing speech access technology, such as Automatic Speech Recognition. In this section, we describe how we introduced an optional ASR component to complement and potentially enhance the navigation functionalities of the Meeting Miner.

7.6.1 Speech Recognition Component

The Meeting Miner was modified so as to take as input individual participants' speech transcripts, in addition to the XML meeting interaction metadata, participants audio profiles and audio files. We used an of-the-shelf speech recogniser to produce the transcripts: the Open Source Sphinx-4 speech recogniser (Walker et al. 2004). We used the standard ASR metric of Word Error Rate (WER), detailed below to describe the quality of the transcripts.

Definition 7.6 Word Error Rate (WER)

Let $Words(s_i)$ represent the exact words of a the spoken sentence s_i, with s_i belonging to the set of speech segments $\mathcal{S}$.

Let $Trans(s_i)$ be the corresponding ASR transcript of sentence s_i.

Let $Diff(String_1, String_2)$ be the function which returns the number of differences (insertion, deletion and substitution) between $String_1$ and $String_2$.

Then the Word Error Rate (WER) is defined as the ratio of the number of insertion, deletion and substitution errors between the speech recognition transcript $Trans(s_i)$ and the exact spoken sentence $Words(s_i)$ over the number of terms in the spoken sentence $Words(s_i)$.

$$WER = \frac{Diff(\ Trans(s_i)\ ,\ Words(s_i)\ )}{|\ Words(s_i)|}$$

As this definition can be difficult to interpret, let us illustrate with the following example how we evaluated the WER of the ASR transcripts of our meeting recordings in practice :

Example 7.1 *Calculating the Word Error Rate:*

Spoken sentence $Words(s_i)$: *"If we pay them that then ... we've got like ... we've paid our costs..."*

ASR transcript $Trans(s_i)$: *"If we pay them that's been ... we've looked like ... we stayed up costs..."*

Number of words in exact sentence $|\ Words(s_i)\ |$: *=15*

Number of Substitutions : *= 4 :"been"* for *"then", "looked"* for *"got", "stayed"* for *"paid"* and *"up"* for *"our"*

Number of Insertions : *= 1 :"'s"* (after *that)*

Number of Deletions : *= 1 :"'ve"* (after second *we)*

$$WER = \frac{4+1+1}{15} = \frac{6}{15} = 0.4 = 40\%$$

For a variety of reasons such as poor recording conditions, noise (participants typing on the keyboard), inappropriate language models, speakers' accents, etc. we obtained very poor recognition results in our meeting recording transcripts . The following Table 7.1 summarises the WER for a subset of 5 meetings (these will be used throughout the rest of this thesis for evaluation purposes and are detailed in Table 8.1). With an average of 60.2%, the ASR transcription result is

Table 7.1: Word Error Rate obtained for the transcripts of meeting evaluation subset

Meeting	WER (%)
A	59.7
B	69.4
C	61.2
D	53.4
E	57.3
Avg. WER	60.2

rather poor: meaning that on average, a misrecognition takes places for roughly two out of every three spoken words.

Rather than trying to increase the recognition results by improving the recording conditions, using more appropriate language models or training the acoustic models to individual participants' voices, we decided to use the transcripts as they were for the following reasons: we envisage a lightweight meeting capture and browsing environment which (i) can be used in a variety of conditions and surroundings (busy office) (ii) without any restriction and language or topics used (iii) which any participant can join at any time. In fact, poor (but not deliberately poor) recognition results corresponded to a possible real-life scenario. Thus, it was judged interesting to see how the system would cope with poor ASR results and how interaction-based browsing would compensate for the ASR shortcomings.

7.6.2 Integrating ASR transcripts in the Meeting Miner

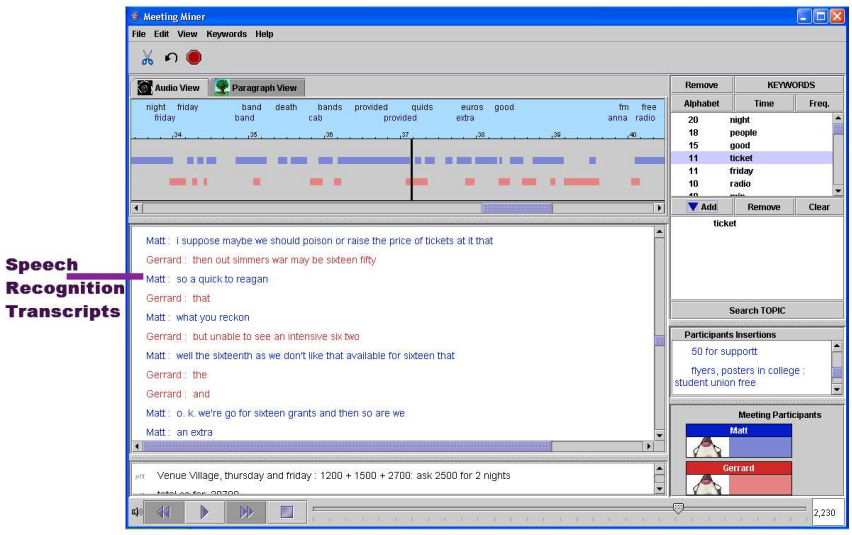


Figure 7.22: Earlier version of Meeting Miner including Speech Recognition transcripts

Once ASR transcripts are available, the difficulty then becomes to incorporate them in the user interface. Speech transcripts can quickly become very substantial in size. Appendix B shows the transcripts of a meeting which involved two participants and last about an hour: these are

nearly 24 A4 pages long! Figure 7.22 illustrates an attempt at including the transcripts in the Meeting Miner GUI. Participants' speech turns were color-coded and time-aligned with the audio file. However, in practice, users of the browser found the transcripts results so poor (as illustrated by Figure 7.23) so as to be useless and even distracting (see user-centered evaluation chapter 9). As a result, the transcripts were removed from the GUI but incorporated into the keyword and topic search functionalities previously described in this chapter.

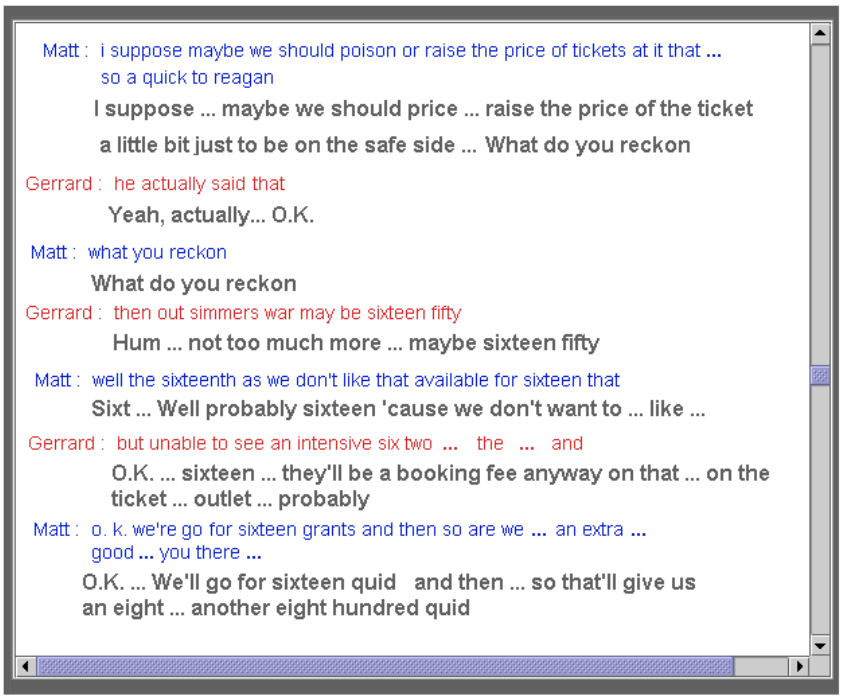


Figure 7.23: Excerpt of participants ASR transcripts (in color) and the corresponding true speech (in grey)

Using both speech transcripts and the content of interaction metadata, search are now prioritised according to some recognition confidence level. Co-occurrence of a term in a text and speech event is assigned the maximum confidence level. The next confidence level is assigned to a term in a text event only (as the analytic evaluation performed in chapter 8 shows greater precision for terms found in text rather than speech events). Finally, the lowest confidence level is assigned to terms found in speech events alone (which corresponds in fact to the WER). In order to obtain reasonably reliable results, we only kept terms of the ASR transcripts which were also found in the meeting Controlled Vocabulary $\mathcal{V}$. They were three main reasons for this: (i) as previously stated, the assumption that the most important meeting keywords would be found in the meeting outcome $\mathcal{O}$, (ii) the intuition that if words are present in $\mathcal{O}$, they are more likely to have been uttered at some stage during the meeting, (iii) trials using all words contained in the transcripts showed that if the system fails repeatedly, users will stop relying on the search functionalities of the system and will quickly revert to browsing the recording through random access, thus defeating

the whole purpose of the meeting browsing tool. As a result of simply limiting the ASR transcripts terms to the ones found in $\mathcal{V}$, the average WER for evaluation subset was immediately reduced from 60.2%. to 39.4%, an immediate reduction of more than 20%. This meant that terms found using speech events alone (lowest confidence condition) failed once out of every three attempts (poor reliability) instead of twice out of every three attempts (not ideal but nevertheless improved reliability).

7.7 Browsing a Meeting with the Meeting Miner

Before moving to the evaluation phase of the browser, we will first illustrate possible meeting recording navigation with the Meeting Miner with a few examples. Interaction information used by the Meeting Miner offers users a large choice of granularities and modalities of retrieval. If a user does not exactly know what he is looking for, a general paragraph neighbourhood retrieval, where all time-based media intervals and operations related to a specific paragraph are retrieved may be appropriate, or, an action-based navigation 7.3.5. However, when the user is more familiar with the meeting content and is looking for specific information, keyword or topic search may be in this case more appropriate.

7.7.1 Browsing with a Paragraph Temporal Neighbourhood

The following example illustrates paragraph temporal neighbourhood browsing. Consider the following two contiguous paragraphs in the final textual outcome of the meeting:

(par4.1) *budget of 3000 from the student union*

(par4.1.1) *maybe charge people more?*

Although this can not be inferred from solely looking at the meeting textual outcome, these paragraphs were modified in an arbitrary order. Before the participants came to the conclusion that they needed to charge people more, in paragraph 4.1.1, they first made changes to a number of paragraphs, in the following order :

(par12.2) *bus hire 1500 for 60 people*

(par4.2) *travel 1500*

(par6.1.1) *4400 for students 2700 for staff in single room*

The subjects participants discuss while these paragraphs are modified are as follow :

(Au1) *(par12.2,4.2) travel arrangements, cost of hiring a bus, existence of a budget*

(Au2) *(par6.1.1) hotel expenses*

(Au3) *(par4.1.1) need to charge people more*

In other words, only when the participants realised that the cost of travel and the hotel would exceed their initial budget (paragraph 4.1) did they make the suggestion that they will need to ask people to financially contribute to the trip (paragraph 4.1.1).

7.7.2 Action-based Navigation

In action-based navigation, it is envisaged that the user will move the slider along the timeline until he sees an information of interest. In Fig. 7.20, the Action Panel highlights the fact that at this specific point in time, one of the participants typed the words: "*€1500 for 60 people*". If the user were to play the corresponding audio, this is what he would hear:

(participant 1) *Ok, then we can get everybody on the bus so I guess we should make everybody go on the bus unless they really want to get there on their own (...) if we are going to need to hire a bus* ***for 60 people*** *anyway*

(participant 2) *That's true but what if they insist that they want to get there by their car*

(participant 1) *Well I guess we'll just have to let them go, I don't know*

(participant 2) *So we'll say then the bus is going to cost* ***€1500*** *for 45 people*

7.7.3 Keyword and Topic Searches

Keyword and topic searches will provide results similar to the previous example, except that the set of speech turns retrieved are not only punctual but will often be spread over the entire meeting duration (whenever the keywords are typed or pointed at). The skip forward/backward functionalities of the Audio Panel can be used to bring the user to the next/previous set of audio segments which have been identified as being relevant to the search. To avoid jitters, segments in close time proximity are merged for smooth listening when the set of audio segments are played back. If a user were to find a specific set of audio segments to be particularly interesting, the audio functionalities provided by the Meeting Miner would enable him to explore surrounding segments.

7.8 Conclusion

In this chapter, we have described in detail how we used interaction metadata to provide access to meeting recordings. We have first presented the necessary underlying concepts (meeting representation, semantic overlap) and assumptions (content based by proxy, link between time-distance semantic relatedness, etc.). We have described how we used these concepts in order to implement keyword and topic search based on interaction events. We have described alternative navigation modalities such as action-based navigation or paragraph temporal neighbourhood retrieval.

Interaction-based navigation seems appropriate for quick visual scanning, as actions will generally have strong associated semantics and potentially offer a powerful indexing method into the time-based media. Paragraph temporal neighbourhood retrieval can potentially uncover semantic links between non-contiguous text items which are not necessarily obvious when solely looking at a meeting's outcome. The appeal of the indexing and navigation methods presented in this chapter lie in the fact that participants' interactions are discrete, generally sparse enough so as not to overload a user with information, and tend to form natural semantic clusters over time (when a specific topic is discussed) allowing for discrimination and segmentation of topics within a meeting recording. We have described how we integrated ASR transcripts in the Meeting Miner, in order to complement the interaction-based retrieval model, and to expand meeting information access. Finally, we have illustrated the system's navigation functionalities with several examples. In the following two chapters of this thesis, we will present an analytic evaluation of the interaction-based search functionalities described in this chapter (chapter 8) followed by a user-centered evaluation chapter (chapter 9).

Chapter 8

Meeting Miner Analytic Evaluation

8.1 Introduction

The main difficulty of an information retrieval system does not lie in it's implementation but in it's *evaluation* (van Rijsbergen 1979). As we have seen in Chapter 2, evaluation of meeting browsing systems have generally been performed on an *"ad-hoc"* basis, depending on the browser's designers specific research interests. Recently, Wellner et al. (2005) have proposed a Browser Evaluation Test (BET), which methodology is general enough so as to be used to evaluate any type of meeting browsing systems. We have used the BET framework in order to perform a similar task-oriented evaluation of the Meeting Miner, which we describe in the next Chapter 9.

In this chapter, we present an analytic evaluation of the keyword and topic search functionalities of the Meeting Miner described in the previous chapter. By *"analytic"* evaluation, we mean an evaluation which does not involve human participants in the information retrieval experiment, but rather relies on statistical methods and measurements, such as the standard IR metrics of precision and recall. In particular we whish to measure to what degree certain assumptions underlying the time-based search mechanisms are true: do participants utter certain words shortly before or after they type them or point at them? Are co-occurrence of certain words in close time proximity indicative of a semantic link between these terms? We first describe the metrics used throughout the evaluation, the evaluation subset and the methodology followed (random sample of terms). Statistical tools and definitions are introduced when necessary in order to ensure the soundness of the results presented and to quantify the uncertainty in the values (margin of errors). Evaluation is performed on the time-based only search functionalities and also on time-based and ASR techniques combined, in order to demonstrate that, and quantify how time-based techniques can be used both as an alternative or a complement to ASR technology. Whenever necessary, we try to illustrate

quantitative results with examples extracted from meeting recordings or reference to other relevant (psycho-linguistic) studies.

8.2 Methodology

8.2.1 Definition of Evaluation Metrics: Precision and Recall

In order to provide an analytic evaluation of the Meeting Miner, we will use the standard information retrieval metrics of precision and recall (van Rijsbergen 1979). Remember from the definition 7.3 in the previous chapter that a *keyword search* is a function $\mathcal{KS} : \mathcal{V} \rightarrow \mathcal{S}$

$$\mathcal{KS}(\mathcal{K}w) = \{s_i \mid (s_i \ overlaps \ T_j) \wedge (\mathcal{K}w \in T_j)\}$$

where $\mathcal{T} = \{T_1, ..., T_{|T|}\}$ is a set of text events (time intervals) and $\mathcal{S} = \{s_1, ..., s_{|S|}\}$ a set of speech segments and $\mathcal{K}w$, a specific keyword. Terms(T_i) is the set of terms contained in the text action (T_i), while Terms(s_j) is the set of terms uttered during the speech interval s_j.

In the context of meeting recordings, we will adapt the definition of the precision and recall metrics as follows:

Definition 8.1 Precision

$$\pi = \frac{\mid \{ \ s_i \mid s_i \in KS(Kw) \wedge Kw \in Terms(s_i) \ \} \mid}{\mid KS(Kw) \mid}$$

This equation defines precision as the ratio of speech segments s_i returned by the keyword search $KS(\mathcal{K}w)$ which actually contain the keyword $\mathcal{K}w$ over all the speech segments s_i returned by the keyword search (whether or not they contain the keyword $\mathcal{K}w$).

Definition 8.2 Recall

$$\rho = \frac{\mid \{ \ s_i \mid s_i \in KS(Kw) \wedge Kw \in Terms(s_i) \ \} \mid}{\mid \{ \ s_j \mid Kw \in Terms(s_j) \ \} \mid}$$

This equation defines recall as the ratio of speech segments s_i returned by the keyword search $KS(\mathcal{K}w)$ which actually contain the keyword $\mathcal{K}w$ over all utterances of $\mathcal{K}w$ in the set $\mathcal{S}$ of all speech segments contained in the meeting.

In addition to measuring precision and recall, we will also use the statistical metric of Standard Deviation (σ), which is useful to measure the *dispersion* of the results across the various keywords. The reason for this, as we will see, is that a reasonably good performance on average can also hide

some occasional very poor results. So in addition to overall performance, standard deviation can thus give us some insight on the *reliability* of the search functionality of the system across a large number of situations and terms.

The standard deviation is calculated as the root of the mean square deviation from the mean of the random variable X. In the following definition, the mean is represented by the notation $\mathcal{E}(X)$ which stands for *expected value* of X.

Definition 8.3 Standard Deviation

If X is a random variable and $\mathcal{E}(X)$ the expected value of X, then the Standard Deviation is calculated as follows:

$$\sigma \quad = \quad \sqrt{\mathcal{E}\,(\ (\ X\ - \mathcal{E}(X)\ )^2\ )}$$

If the random variable X takes the values $(X_1, ..., X_N)$ with equal probability, then the Standard Deviation is calculated as follows:

$$\sigma \quad = \quad \sqrt{\frac{1}{N}\sum_{i=1}^{N}(X_i - \overline{X})^2}$$

Where $\overline{X} = \frac{1}{N}\sum_{i=1}^{N} X_i$ is the mean of X.

8.2.2 Meeting Evaluation Subset and the Random Samples' Approach

Given the previous definitions, calculating precision and recall of the Meeting Miner search functionality on our meeting corpus would involve for every single keyword Kw:

- listening to the set of all speech segments returned by the keyword search $KS(\mathcal{K}w)$ and counting those which contain the specified keyword (precision)
- subsequently, listen to the whole meeting and counting every single occurrence of the specified keyword (recall).

It should be obvious by now that this task is extremely time consuming. Therefore, performing this operation for every single keyword in a meeting, which may contain hundreds of keywords and could be more than an hour long, for every meeting in our corpus would clearly be intractable. As an alternative, we measure precision and recall on a significant sample of keywords from a subset of our meeting corpus. The reader will remember from the definition 7.2 that the Meeting Miner uses the notion of a Controlled Vocabulary $\mathcal{V}$, consisting of the words typed during the meeting (less stop words). In order to ensure the *significance* of our evaluation sample, we impose the following constraints:

- **Size** The size of the sample must be significant in comparison the overall Vocabulary size (i.e. at least 20 % of the Controlled Vocabulary $\mathcal{V}$)

- **Representative** The sample should cater for possible bias due to the Written Term Frequency of the keywords.

As a result of these consideration, the keywords sample for each meeting is chosen as follows:

Composition of keywords' sample for the evaluation of precision & recall
... for every single meeting in the evaluation subset
→ pick the 5 keywords with highest Written Term Frequency in $\mathcal{V}$
→ pick a random sample of 5 of the lowest Written Term Frequency words (typically 1) in $\mathcal{V}$
→ pick a random sample of 10 keywords in $\mathcal{V}$ with random Written Term Frequency

Therefore, the sample of random keywords is of size 20 per meeting and we perform the same operation over 5 meetings. Thus, our measurement of precision and recall is based on a random sample of 100 keywords in total, extracted from five different meetings of our corpus. Table 8.1 summarises the information about the meetings picked for the evaluation purpose in terms of size of controlled vocabulary and duration of meeting recording. As the total size of the controlled vocabulary is 450 terms, our sample's size of a 100 represents: 100 / 450 = 22.2 % of the total number of keywords in the controlled vocabulary of the five meetings.

Table 8.1: Description of meeting evaluation subset

Meeting ID	Number of Written Keywords in Meeting	Meeting Duration
A	116	55 min - 44 s.
B	140	1h - 20 min
C	48	37 min - 32 s.
D	61	47 min - 57 s.
E	85	44 min - 40 s.
TOTAL	450	4 h. - 25 min - 53 s.

8.3 Precision & Recall of Time-based Keyword Search

The results in this section describe evaluation of precision & recall for the *time-based only* keyword search, while the search functionality using time-based and ASR techniques *combined* is described and evaluated in the following section 8.4.

8.3.1 Highest Written Frequency Keywords

Table 8.2 shows the result of the analytic evaluation performed on the highest ranking keywords. In the table, WTF stands for Written Term Frequency, which corresponds to the number of time the keyword appears in the meeting final text document $\mathcal{O}$, STF stands for Spoken Term Frequency,

which stands for the amount of time the corresponding keyword is uttered during the meeting and the Avg. Time corresponds to the average listening time required within the speech segments s_i returned by the keyword search $KS(\mathcal{K}w)$ before the keyword Kw is actually heard (if indeed it is present). The results for the sample of 20 highest keywords within 5 meetings of our corpus are quite remarkable: the precision value of .898 shows that in 89.8% of the cases, a speech segment s_i returned by the Meeting Miner search functionality does contain the specified keyword and that it takes on average 11.2 seconds of listening time to hear the given keyword. The high precision result would seem to confirm Assumption 7.1 formulated in the previous chapter which stated that when users type or point at a word in the meeting document, they are indeed very likely to have uttered this word in close time proximity. Recall stands at 57.3%, which means that overall, the Meeting Miner keyword search retrieves slightly more than half of all utterances of a given keyword for the terms with highest WTF.

Table 8.2: Precision, Recall and average listening time of keyword search for Highest Written Frequency Keywords

Meeting	Keyword	WTF	STF	Avg. Time (in s.)	Precision	Recall
A	*Band*	9	45	14.4	0.833	0.4
A	*Ticket*	7	17	8.2	0.857	0.412
A	*Cost*	6	24	11.2	1	0.375
A	*Capacity*	6	5	12.6	0.75	0.6
A	*Night*	5	37	9.5	1	0.432
B	*Night*	24	52	10.4	1	0.865
B	*Fly*	16	44	9.3	0.875	0.341
B	*Train*	13	30	11.3	1	0.767
B	*Hong Kong*	13	32	11.3	1	0.656
B	*Shangai*	11	38	10.3	0.667	0.447
C	*Hotel*	6	7	14	0.833	0.857
C	*Student*	5	11	4	0.545	0.5
C	*Staff*	5	7	15.2	1	0.427
C	*Room*	4	9	14	0.67	0.375
C	*Travel*	3	2	17	0.67	1
D	*Price*	4	26	6.2	0.75	0.529
D	*Second*	4	8	22.2	1	0.5
D	*Team*	3	10	9	1	0.9
D	*Student*	3	12	17.1	1	0.667
D	*Hamper*	3	4	14.5	1	0.5
E	*Dublin*	6	8	6.4	1	0.75
E	*Cottage*	6	13	8.5	1	0.615
E	*Sea*	5	12	11.2	1	0.583
E	*House*	4	19	8.2	1	0.316
E	*New Ross*	4	6	5	1	0.5
Sum - Σ		**175**	**478**			
Mean - $\mathcal{E}(X)$				**11.2**	**0.898**	**0.573**
Std. Dev. - σ				**4.2**	**0.143**	**0.191**

8.3.2 Least Written Frequency Keywords

Table 8.3 contains the results obtained for the random sample of least written frequency terms. The overall precision result (89.8%) is slightly higher than the one obtained for the highest Written Term Frequency keywords, thus reinforces Assumption 7.1 about the proximity of keyword in text events and speech turns. More surprising is the result for recall: the overall result is 60%. This is truly remarkable and unexpected as with a Written Term Frequency equal to 1 for these keywords, the search functionality based on a text event can only point to one *single location* in the meeting recording. The ratio of written to spoken term frequency stands at: WTF/STF = 25/151 = 16.6. In other words, although the frequency of text events containing a specific keyword represents only 16.6% of the corresponding keyword's Spoken Term Frequency, these are sufficient to map 60% of the spoken utterances.

Table 8.3: Precision, Recall and average listening time of keyword search for Least Written Frequency Keywords

Meeting	Keyword	WTF	STF	Time (s.)	Precision	Recall
A	*Crew*	1	1	32	1	1
A	*Free*	1	4	22	1	0.25
A	*Expect*	1	12	7	1	0.5
A	*Sunday*	1	6	28	1	0.5
A	*Nottingham*	1	1	27	1	1
B	*Start*	1	6	1	1	0.667
B	*Yunnan*	1	30	2	1	0.033
B	*Museum*	1	1	16	1	1
B	*City*	1	25	8	1	0.12
B	*Shopping*	1	1	2	1	1
C	*City*	1	2	11	1	0.5
C	*Budget*	1	3	18	1	1
C	*Week-End*	1	6	5	1	0.167
C	*Saturday*	1	9	-	0	0
C	*Subsidised*	1	2	1	1	0.5
D	*Flyers*	1	3	8	1	1
D	*Table*	1	4	5	1	0.667
D	*Equipment*	1	1	5	1	1
D	*Budget*	1	1	7	1	1
D	*People*	1	7	3	1	0.167
E	*Glenbeg*	1	2	4	1	0.5
E	*Town*	1	14	3	1	0.429
E	*River*	1	2	4	1	1
E	*Train*	1	4	-	0	0
E	*People*	1	4	24	1	1
Sum - Σ		**25**	**151**			
Mean - $\mathcal{E}(X)$				**10.6**	**0.92**	**0.6**
Std. Dev. - σ				**9.5**	**0.271**	**0.372**

In order to understand why recall is surprisingly high in this case, one needs to take a closer look at the set of segments returned by the Meeting Miner search functionality and analyse the corresponding speech turns. Close inspection of these segments indicated that the answer to the

unusual high recall of our system can be explained by a number of psycholinguistic phenomena. In many cases, the use of certain keywords were often (but not always) extremely localised. A distinctive pattern was identified: when a participant started to use a specific keyword, other participants would naturally and immediately start to repeat the same word (*linguistic convergence*).

Close repetition of terms in discourse Tannen (1989) can take various forms: repetition of one's own words (self-repetition) vs. repetition of other peoples' words (allo-repetition), exact words vs. paraphrase, repetition with slight variations, syntactic repetition, also known as *priming* (Pickering and Branigan 1999, Smith and Wheeldon 2001), which in this case consists in reproducing a sentence's structure rather than content, etc. The type of repetitions encountered in our meeting corpus which have clear implications for the recall performance of the text-event-to-speech search functionality of the Meeting Miner system are repetitions of the *exact* same term in very close time proximity, which is sometimes referred to as "linguistic convergence" or automatic *"shadowing"*. Automatic shadowing is described by Tannen (1989) as: *"repeating what is being heard with a split-second delay"*. Repetition is a natural and pervasive phenomenon in discourse and can be attributed to a number of factors such as fluency of speech (reduced latency), increased comprehension, emphasis, acknowledgement, discourse cohesion and coherence, etc. Exact term repetition is a very distinctive pattern in our online meeting recordings: participants will often bounce a term off each other for a few sentences, and these very words are more likely to be the ones written down. Close term repetition is illustrated by the following examples extracted from our corpus:

Example 8.1 *Term Repetitions in close time proximity*

(participant 1) Nady... in New Ross ***town*** *as well, so that's in New Ross... I'd say the good thing about these places is that they're probably going to be handy to get to*

(participant 2) Because it's in New Ross, which is a fairly big ***town*** *?*

(participant 1) Yeah, I'd say it'll be ... it's going to be straightforward to get there...

(participant 2) That's funny, it's kind of in a residential area...

(participant 1) Yes, because it is in the ***town***

(participant 2) So it's in ***town*** *then...*

(participant 1) in the ***town****, in* ***town***

An other example of term repetitions is illustrated by the example 8.2.

Example 8.2 *Term Repetitions*

(participant 1) ... Maximum ***expected*** *... so for* ***expected*** *should we say ... because seven hundred seems like a big gig...*

(participant 2) Hum ... I guess so, yeah ... Well, let's say ***expected****, I would say between four hundred and five hundred*

(participant 1) O.K. let's say then four hundred and fifty ***expected****... If things go well, we'll go for seven hundred and plus ... so that's the* **number of people** ***expected*** *to be there.*

(participant 2) So... O.K., we've got the band and the **number of people** *we're* ***expecting****...*

Participant 1 then wrote: "expected: 450". A final example of linguistic convergence is illustrated by the example 8.3, whereby the combination of the words used by the two participants is the one which is eventually jolted down:

Example 8.3 *Linguistic Convergence*

(participant 1) ...I've been to a couple of gigs there and the **sound** *wasn't very good there... I found the* **sound***...*

(participant 2) So the **sound quality** *itself then ?*

(participant 1) Yeah, the **sound quality wasn't great**

The participants eventually jolted down: **"sound quality wasn't great"**.

The phenomena of automatic shadowing and term repetitions thus explain why even very few text events may be sufficient to uncover a significant number of identical terms in a "keyword cluster". Similarly, if a single word is repeated on several occasions, when it comes to writing down some notes, then it is reasonable to assume that the participants are also more likely to write down the one which is currently in use (linguistic convergence) as opposed to another synonym.

Although the overall recall result for the least frequency terms can be interpreted as reasonably good, and certainly above expectations, the figure nevertheless also hides some very poor performance as evidenced by a very high standard deviation of $\sigma = 37.2\%$. While the presence of

keyword clusters is a prominent pattern throughout the meetings' recordings, this does not imply that the use of all terms is systematically temporally confined. Many terms are also often used throughout the whole course of the meetings. In Table 8.3, the term *"Yunnan"* is said 30 times throughout the duration of meeting B, but yet, it is only written once. In this case, recall is a paltry 3.3 %. Other examples of keywords with poor recall result (*"City"*, *"Saturday"*, etc) confirms that using text events is therefore insufficient to insure the completeness of keyword spotting in all cases. These results are however to be expected: while the high precision of the keyword vindicates Assumption 7.1 (a participant is likely to utter or have uttered a word if he typed it), the converse assumption is intuitively false: because a participant utters a word by no means signifies that he should necessarily type it immediately afterwards. Thus, in order to increase recall, the complementary technology of Automatic Speech Recognition seems in this case definitely warranted.

8.3.3 Random Written Frequency Keywords

Precision and recall results for our last sample of 50 keywords of random frequency over the 5 meetings subset are presented in Table 8.4. The overall precision stands at 78% and recall at 47.5% . While still reasonably good, these results are less than what we obtained for the previous two samples. If we take a closer look at the results in Table 8.4, one will notice that the results were particularly poor for the third meeting C. The explanation for this beneath average performance is twofold. On the one hand, Table 8.1 shows that this meeting is the one with the least keywords in it's vocabulary, which in turn is a result of text events in this meeting being very sparse. The other factor to take in consideration as regards this particular meeting is that there was also a higher latency between participants speech turns and text events and as a result, the search functionality of the Meeting Miner with it's default overlap value (see 7.1) was ill-suited for this particular type of interaction. In such circumstances, increasing the Overlap Tolerance Value would also increase the performance of the system. However, the weaker performance of the system exposed by meeting C highlights the fact that in scenarios where very little actions are performed during the meetings, the Meeting Miner offers little or no benefit over a pure ASR-based system.

Table 8.4: Precision, Recall and average listening time for 50 random keywords over five samples of the meetings corpus

Meeting	Keyword	WTF	STF	Avg. Time (s.)	Precision	Recall
A	*Promotion*	4	2	26	0.333	0.5
A	*Play*	4	16	18.3	1	0.625
A	*Village*	3	13	28.7	1	0.462
A	*Price*	3	8	2	0.333	0.25
A	*Cons*	3	5	4	1	0.8
A	*Week*	3	7	13	1	0.571
A	*Venue*	3	23	19	1	0.348
A	*Polar*	2	6	18.5	1	0.5
A	*Stage*	2	4	4	1	0.75
A	*Provided*	2	2	1	1	1
B	*Wall*	7	33	16.6	1	0.303
B	*Hire*	6	11	16.3	1	0.818
B	*Ping Yao*	5	21	11.3	1	0.429
B	*Journey*	4	6	22	0.333	0.167
B	*Leave*	3	5	13.5	1	0.6
B	*Arrival*	3	5	5	1	0.6
B	*Palace*	3	17	14.5	0.667	0.235
B	*Hotel*	2	2	18	1	1
B	*Dynasty*	2	2	-	0	0
B	*Kunmig*	2	9	5	1	0.556
C	*NewPark*	3	2	13	0.333	0.5
C	*Racing*	3	6	5	0.5	0.167
C	*Person*	3	5	0	0	0
C	*Greyhound*	3	6	5	0.5	0.167
C	*Afternoon*	2	5	1	0.5	0.2
C	*Kilkenny*	2	6	10	0.5	0.167
C	*House*	2	1	-	0	0
C	*Staying*	2	2	4	0.5	0.5
C	*Music*	2	3	3	0.5	0.333
C	*Cost*	2	6	5	0.5	0.167
D	*Drinks*	3	4	10	0.5	0.5
D	*Union*	3	6	7.33	1	0.667
D	*Euro*	3	10	8	1	0.4
D	*Quizz*	2	4	5	1	0.75
D	*Raffle*	2	7	4.67	1	0.857
D	*Questions*	2	10	14.5	1	0.3
D	*Print*	2	4	6	1	1
D	*Tickets*	2	12	4.5	1	0.3
D	*Free*	2	1	16	1	1
D	*Number*	2	2	7	1	1
E	*Near*	4	18	14	1	0.333
E	*Dungarvan*	3	12	3	1	0.417
E	*Booked*	3	12	8.33	1	0.833
E	*Brookvale*	3	3	21	1	1
E	*Remote*	2	4	14	1	0.25
E	*Thursday*	2	2	-	0	0
E	*Choice*	2	4	3	1	0.5
E	*Inland*	2	4	7	1	0.5
E	*Village*	2	6	5	1	0.167
E	*Beach*	2	11	5	1	0.273
Sum - Σ		**138**	**375**			
Mean - $\mathcal{E}(X)$				**9.9**	**0.78**	**0.475**
Std. Dev. - σ				**7**	**0.334**	**0.297**

8.3.4 Results: All Keywords

Table 8.5 summarises the results for precision and recall over the sample of a 100 keywords of random frequency over the 5 meetings subset and will be the one used for the overall analytic evaluation of the system, *without* the inclusion of ASR technology. The keywords sample evaluated was of significant size, representing 22% of all terms in the joint controlled vocabulary $\mathcal{V}$ of the 5 meetings, which translates into 338 written occurrences (WTF) and 1004 spoken occurrences (STF). In order to give a measure of confidence in the results, we use the statistical metric of Standard Error:

Definition 8.4 Standard Error

If $\mathcal{E}(X)$ is the Expected Value of the random variable X among a sample of the total population and **n** *the size of the sample used to determine this estimation then the Standard Error is defined as:*

$$\textit{Std. Err. (X)} \quad = \quad \sqrt{\frac{\mathcal{E}(X)\ (1-\mathcal{E}(X))}{n}}$$

When the sample taken for calculating the Expected Value $\mathcal{E}(X)$ constitutes a significant proportion of the total population, as in our case (22.2%), a *Finite Population Correction* factor can be applied as follows:

Equation 8.1 Standard Error with Finite Population Correction

If $\mathcal{E}(X)$ is the Expected Value of the random variable X among a sample of the total population of size **N** *and* **n** *the size of the sample used to determine this estimation then the Standard Error:*

$$\textit{Std.Err.}(X) \ = \ \sqrt{\frac{\mathcal{E}(X)\ (1-\mathcal{E}(X))}{n}} \ \cdot \sqrt{\frac{N-n}{N}}$$

In our case, since the sample represent 22.2% of the total number of terms, the finite population correction factor is equal to: $\sqrt{1-.222} = .882$. In order to guarantee that our results are correct with a confidence level of 95%, a confidence interval is calculated as our expected value plus or minus twice the Standard Error:

Definition 8.5 Confidence Interval

For a confidence level of 95%, the Confidence Interval, given an Expected Value $\mathcal{E}(X)$ is:

$$\textit{Conf. Interval} \quad = [\ \mathcal{E}(X) - 2.\textit{Std.Err.}(X)\ ,\ \mathcal{E}(X) + 2.\textit{Std.Err.}(X)\ ]$$

Table 8.5: Precision, Recall and average listening time for 100 random keywords of random frequency over five samples of the meetings corpus

	WTF	STF	Avg. Time (s.)	Precision	Recall
Sum - Σ	**338**	**1004**			
Mean - $\mathcal{E}(X)$			**10.1**	**0.844**	**0.531**
Std. Dev. - σ			**7.3**	**0.288**	**0.301**
Std. Err.				**3.2%**	**4.4%**
Conf. Interval				**+/-(2x) 3.2%**	**+/-(2x) 4.4%**

Overall precision stands at π = 84.4 [+/- (2x) 3.2]% which means that the text-to-speech-based search functionality of the Meeting Miner defined in 7.3 is highly reliable, therefore vindicating the assumption 7.1, while the average listening latency required before hearing the specified keyword is 10.1 seconds. Recall is equal to ρ = 53.1 [+/- (2x) 4.4]% which would indicate that the text-to-speech-based search functionality is overall able to retrieve roughly every second occurrences of a written keyword. While this is somehow insufficient, it is nevertheless encouraging given the sparsity of text events during the meetings. In the sample, the ratio of written frequency to spoken frequency is equal to r = 338/1004 = 33.7% which is considerably less that the value obtained for recall. The higher than expected value obtained for recall can essentially be attributed to the phenomena of repetition in discourse and linguistic convergence between meeting participants as discussed above.

8.4 Precision & Recall of Time-based and ASR combined Keyword Search

The previous section demonstrated that time-based techniques for word spotting achieve very good precision results. Recall, while higher than we had anticipated, is far from being entirely satisfactory. At 53.1%, it means that we miss on average every second occurrence of every term in our vocabulary. In section 7.6, we have found that the average WER for the transcripts of our evaluation subset was 60.2%. However, using only a subset of words from the controlled vocabulary $\mathcal{V}$, we managed to subsequently reduce the WER to 39.4%, a reduction of more than 20%. In this section we measure how using our speech recognition transcripts for word spotting fares in comparison to the time-based technique evaluated in the previous section. In particular, we investigate how combining the two techniques affect precision and recall.

Figure 8.1 represents all the possible outcomes for a Word Spotting task performed on a meeting recording using the Meeting Miner. It is composed of four sets and all their possible intersections. The sets are: Words($\mathcal{S}$), set of all the spoken words, ($\mathcal{V}$), the subset of spoken words which belong to the controlled vocabulary, Trans($\mathcal{S}$), the set of words extracted from the ASR transcripts and finally TB, the set of words spotted by the time-based search of the Meeting Miner.

The set Words($\mathcal{S}$) represents all the spoken words, that is the entire set of words which could in principle be correctly spotted. As we have a limited Controlled Vocabulary, a significant number of terms uttered during the meeting are Out-of-Controlled-Vocabulary (OCV) terms (OCV = Words($\mathcal{S}$) - $\mathcal{V}$). The next set of words Trans($\mathcal{S}$) represent all the words obtained through the ASR component. A significant number of terms (WER) in Trans($\mathcal{S}$) are misrecognised words. Some of the OCV terms may have possibly been correctly recognised by the ASR component (vertical hash in the figure), but because of the very high Word Error Rate, we decided not to take them into account, privileging high precision and lower recall to very poor precision for a slightly higher recall. Our last set consists of TB, the set of words spotted by the time-based search of the Meeting Miner. We wish to stress that he percentages expressed on the figure represent a proportion of the *total number of words which were actually spotted by the ASR and time-based techniques combined*. They dot not represent a percentage of the possible terms which could have been spotted (recall) and thus should not be interpreted as such! What the figure says is that of all the terms which were correctly spotted by combining the ASR and time-based techniques, 53.7% were spotted exclusively by the time-based search, 22.2% were spotted exclusively by Speech Recognition and 24.1% were both spotted by the time-based search and in the ASR transcripts.

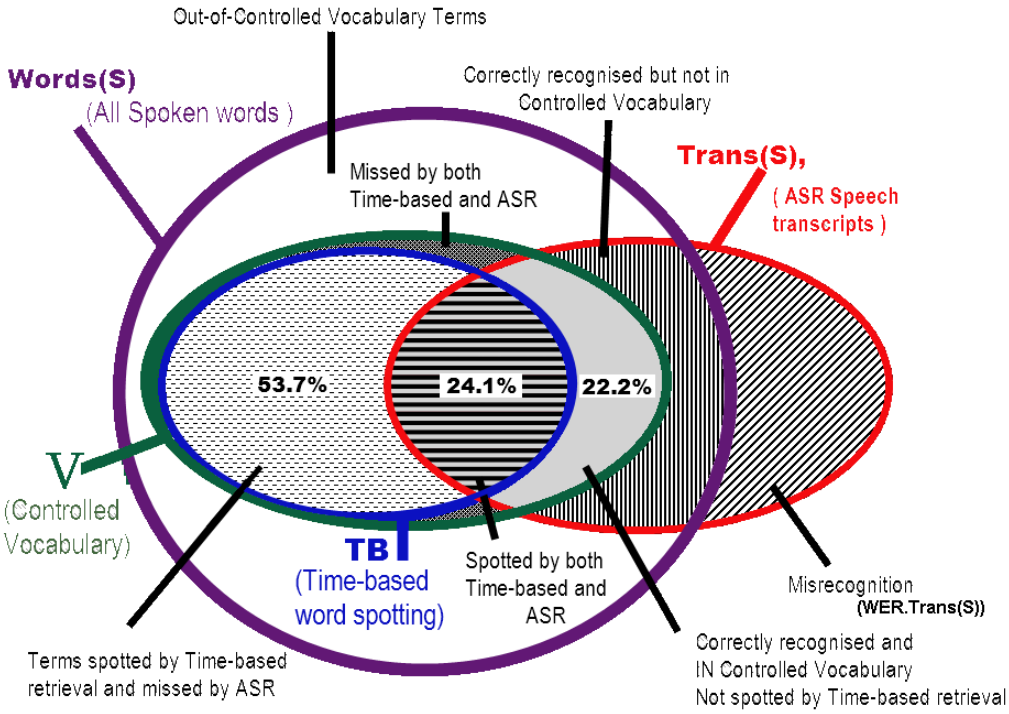


Figure 8.1: Word Spotting Sets, the figures represent a percentage of the combined number of words correctly spotted by the ASR and Time-based techniques

To estimate the loss in precision introduced by the using ASR for Word Spotting, one needs to calculate the error introduced by the WER. Let $\mathcal{C}s$ stand for the number of correctly spotted words. Figure 8.1 shows that 46.3 % of $\mathcal{C}s$ (24.1 + 22.2) are obtained through ASR transcripts.

This corresponds to the number of words correctly recognised: (1-WER) $\times$ $Trans_{\mathcal{V}}(S)$ with a corresponding speech error of ASR(Err) = WER $\times$ $Trans_{\mathcal{V}}(S)$.

$$
\begin{aligned}
\ldots \quad & \text{(1-WER)}.Trans_{\mathcal{V}}(S) = 46.3\ \%\ .\ \mathcal{C}s \quad \text{(divide by (1-WER)...)} \\
\rightarrow \quad & Trans_{\mathcal{V}}(S) = \tfrac{46.3\%}{1-WER}.\mathcal{C}s \quad \text{(substitute in ASR(Err)...)} \\
\rightarrow \quad & ASR(Err) = WER.\ \tfrac{46.3\%}{1-WER}.\mathcal{C}s
\end{aligned}
$$

Now, 78.7 % of $\mathcal{C}s$ (53.7 + 22.2) are obtained through the time-based search TBSearch(), which has a precision of π= 84.4% with a corresponding search error of Search (Err).

$$
\begin{aligned}
\ldots \quad & \pi\ .\ \text{TBSearch()} = 78.7\ \%\ .\ \mathcal{C}s \quad \text{(divide by } \pi\text{...)} \\
\rightarrow \quad & \text{TBSearch()} = \tfrac{78.7\%}{\pi}.\mathcal{C}s \\
\rightarrow \quad & Search(Err) = (1-\pi)\ .\ \text{TBSearch()} \quad \text{(substitute with previous...)} \\
\rightarrow \quad & Search(Err) = (1-\pi).\tfrac{78.7\%}{\pi}.\mathcal{C}s
\end{aligned}
$$

Finally, our precision estimate is equal to the ratio of correct terms spotted (true positive) which is $\mathcal{C}s$ over the true positive and false positive (the combined error of the time-based search Search (Err) and ASR error: ASR(Err)

$$
\begin{aligned}
(1)\ldots \quad & \text{Precision} = \frac{\mathcal{C}s}{\mathcal{C}s+ASR(Err)+Search(Err)} \\
(2) \rightarrow \quad & \text{Precision} = \frac{\mathcal{C}s}{\mathcal{C}s+WER.\ \frac{46.3\%}{1-WER}.\mathcal{C}s+(1-\pi).\frac{78.7\%}{\pi}.\mathcal{C}s} \\
(3) \rightarrow \quad & \text{Precision} = \frac{1}{1+46.3\%.\ \frac{WER}{1-WER}+78.7\%.\frac{1-\pi}{\pi}} \\
(4) \rightarrow \quad & \text{Precision} = \frac{1}{1+.301+.145} \\
(5) \rightarrow \quad & \text{Precision} = 69.2\%
\end{aligned}
$$

Which gives us a precision of 69.2%. One can verify the correctness of the above calculation by looking at step 2: if there was no ASR component than the equation would become:

$$(2) \rightarrow \quad \text{Precision} = \frac{\mathcal{C}s}{\mathcal{C}s + 0.\ \frac{WER}{1-WER}.\mathcal{C}s + 1.\frac{(1-\pi)}{\pi}.\mathcal{C}s}$$

$$(3) \rightarrow \quad \text{Precision} = \frac{\mathcal{C}s}{\mathcal{C}s + .\frac{(1-\pi)}{\pi}.\mathcal{C}s} \quad (\text{... simplify by removing Cs})$$

$$(4) \rightarrow \quad \text{Precision} = \frac{1}{1+\frac{1-\pi}{\pi}} \qquad (\text{... multiply numerator and denominator by } \pi)$$

$$(5) \rightarrow \quad \text{Precision} = \frac{\pi}{\pi+1-\pi}$$

$$(6) \rightarrow \quad \text{Precision} = \pi$$

This gives us the same precision π calculated without the ASR component. Figure 8.2 illustrates the precision performance of the various word spotting techniques: (i) time-based alone, (ii) time-based and ASR combined, (iii) ASR restricted to the controlled Vocabulary $\mathcal{V}$ and (iv) ASR alone. Note that the results are somehow biased towards the time-based technique as the time-based search returns a speech segment as opposed to a single utterance for ASR. In other words, the better performance also comes with a cost: that of listening to a speech turn instead of finding the exact occurrence of a word. As Table 8.5 shows, this cost consists in an average listening time of 10.1 s. before the keyword is actually heard. As meeting recordings may be several hour longs, an average of 10.1 s. listening cost per term would seem reasonable. Another factor to take into account is that it is extremely difficult to understand a sentence when caught in the middle, so even with ASR word-spotting, play-back may still be required to start from the beginning of a sentence for comprehension reasons.

In order to calculate a new estimation of recall which includes ASR-based word spotting, we use an estimated value derived from: (i) the recall score calculated on the time-based keyword search alone and (ii) the combined ASR-TB word spotting scores. As shown in Figure 8.1, we have found that of all words spotted by the Meeting Miner system including ASR word spotting, 22.2% are spotted exclusively by the speech recognition component. This corresponds to an increased performance of: $\frac{22.2}{53.7+24.1} = 28.5\%$ more words spotted than by the time-based technique alone. As we calculated recall to be 53.1% for time-based word-Spotting, we thus infer that our speech recognition component has led to an increase in recall of: $53.1 \times (1 + \frac{22.2}{53.7+24.1}) = 68.3\%$. These results are illustrated in Figure 8.3 and summarised in Table 8.6

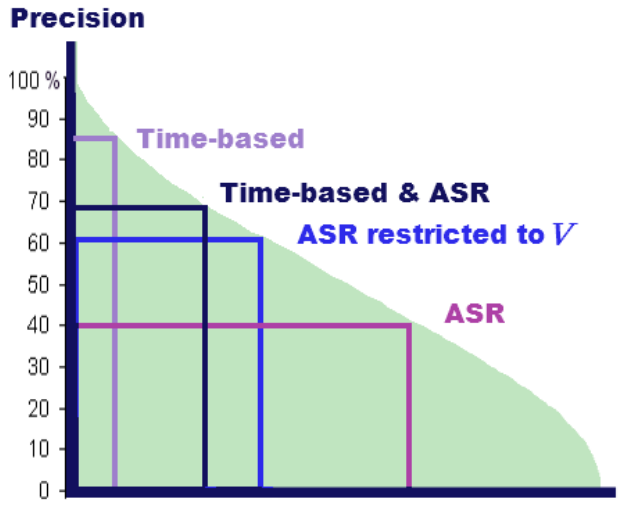


Figure 8.2: Precision results for combinations of Time-based and ASR-based word spotting techniques

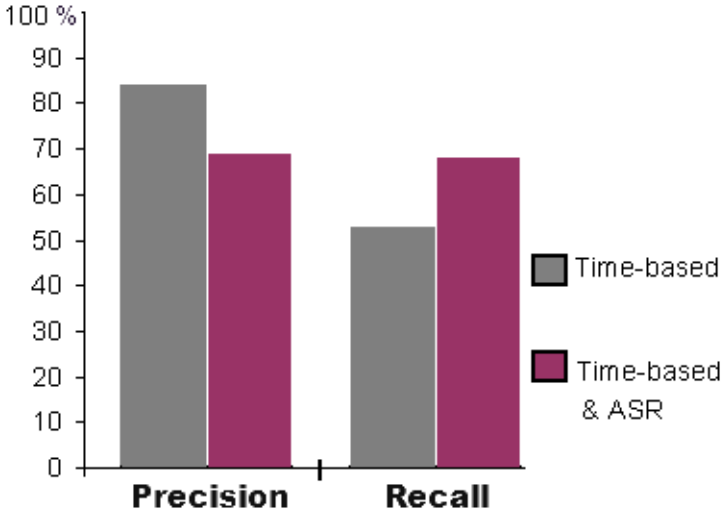


Figure 8.3: Estimation of Precision and Recall for Time-based only and combined ASR and Time-based word spotting (WER=39.4%)

Table 8.6: Estimation of Precision and Recall for Time-based only and combined ASR and Time-based word spotting (WER=39.4%)

	Precision	Recall
Time-based only	84.4	53.1
Time-based & ASR	69.2	68.3
Confidence Interval	**+/-(2x) 3.2%**	**+/-(2x) 5.7%**

8.5 Topic Search Evaluation

Assumption 7.2 formulated the hypothesis that in the complex query $\mathcal{TS}\ (\mathcal{K}w_1 \wedge ... \wedge \mathcal{K}w_j)$, the time distance between the keywords $(\mathcal{K}w_1, ..., \mathcal{K}w_j)$ was inversely proportional to the semantic relatedness likelihood between the keywords. Remember that we did not state that time as a measure of (inverse) likelihood was valid for *any* pair of keywords but applied *only* to terms formulated in a user query. In other words, the keywords $(\mathcal{K}w_1, ..., \mathcal{K}w_j)$ are assigned a semantic *likelihood* given a certain context (e.g. prior knowledge about the meeting, some information expectation or simply common sense) and this likelihood is measured as the distance between occurrences of the terms in the user query. In order to measure the validity of this assumption, we examine the complex queries formulated by the users who participated in the information retrieval task detailed in the next chapter 9.4. We measure the complex query search Performance using the terminology traditionally used for Word-Spotting. If the complex query search $\mathcal{TS}\ (\mathcal{K}w_1 \wedge ... \wedge \mathcal{K}w_j)$ returns a speech segment s_i which contain the keywords $\mathcal{K}w_1, ..., \mathcal{K}w_j$ and these have a shared meaning (i.e. refer to the same topic or object), then this is considered a True Hit. If the search returns speech segments in which the terms are not semantically related (i.e. refer to different topics or objects) then this is considered a False Alarm. As that the previous criteria can somehow seem subjective, we provide the following example in order to illustrate how we distinguish between a True Hit and a False Alarm.

Example 8.4 *Example of shared meaning: True Hit and False Alarm.*

Topic Search: $\mathcal{TS}$ ("**Price**" $\wedge$ "**Ticket**") $= \{s_i\}$

Speech segment s_i	True Hit	False Alarm
"O.K. so we'll set the **price of a ticket** *to 15 quid"*	X	
"...while he went to get the **tickets**, *I decided to go and check out the* **price** *of an Ice-Cream"*		X

If the keywords $\mathcal{K}w_1, ..., \mathcal{K}w_j$ are semantically related in some location of the meeting recording and this is not uncovered by the system, then this is considered a Miss. Finally, if the system can find no match to the query and indeed no match within the recording exists, than this is a considered a *"No Match"* case. A No Match is an important case because it correctly provides to the user the information that no semantic link can be inferred between the query terms within the meeting recording.

The results are presented in the Appendix in Table C.3, which lists the exhaustive combination of terms used as topic searches by subjects in the task-oriented information retrieval experiment described in section 9.4. If we first ignore the miss cases, then of all queries, the Meeting Miner

returns a correct diagnosis in $\frac{61+5}{61+5+7} = 90.4\%$ of cases, either referring to speech turns in which the terms are indeed part of the same topic or alternatively not referring to any speech turns when the combination of keywords were not part of the same topic at any time during the meeting. The level of False Alarm is as low as 9.6%, in which cases co-occurrences of the query terms were incidental. Thus, the very high percentage of speech turns returned by the Meeting Miner in which the close time proximity of keyword co-occurrence translates into a semantic link between these terms vindicates Assumption 7.2. One issue highlighted by Table C.3 is the high number of Miss cases which stand at 38.1%. This can be explained by the fact that text events may be too sparse to detect every co-occurrence of certain terms, while the ASR performance too poor to identify all co-occurrences of the specified keywords in speech (if WER=40%, then the chances that 2 consecutive words $\mathcal{K}w_i$ *and* $\mathcal{K}w_{i+1}$ are *both* recognised correctly is: $\frac{60}{100} \times \frac{60}{100} = 36\%$). Using ASR transcripts with a lower WER, we envisage a reduction in the number of Miss cases and an increase in False Alarms (as more correct cases would be detected at the expense of identifying more incidental spoken term coocurrences), similarly to the pattern identified in 8.4.

8.6 Concluding Remarks on Analytic Evaluation

We conclude the analytic evaluation of the Meeting Miner keyword and topic search functionalities with a few remarks. We have used the information retrieval metrics of precision and recall to evaluate the keyword search, comparing time-based alone and time-based and ASR combined techniques. The high precision (84.4%) [+/- (2x) 3.2%] of the time-based search proved that Assumption 7.1 is correct: in a majority of cases, a typed word has or will also be uttered in close time proximity. Due to the phenomena of repetitions in discourse and linguistic convergence, we found that recall was at (53.1%) [+/- (2x) 4.4%] higher than expected. Combining speech recognition (with WER=39.4% on Controlled Vocabulary) led to an (estimated) increase of recall to 68.3% [+/- (2x) 5.7%] at the expense of reducing precision to 69.2 [+/- (2x) 3.2%]. The topic search functionality using the (inverse) time difference between query terms returned correct results (True Hits & No Match) in 90.4% of cases and False Alarms in only 9.6% of cases. This shows that the inverse time difference between a user's query terms can indeed be used as a very reliable measure of semantic link likelihood (Assumption 7.2).

The fact that the system has shown high precision for keyword searches and high number of True Hits for topic searches has demonstrated that the retrieval system can be used *"reliably"*. The lower recall and the significant rate of Miss in the topic search shows nevertheless that the system can not claim to be *"exhaustive"* and will not necessarily succeed in uncovering all occurrences of keywords and topics in a recording. However, another important factor to take into account is the fact that it is usually easier and faster for participants to talk rather than write during meetings. As such, we will argue that participants interact with the space-based document for

important reasons *during* significant events of a meeting: a plan has been laid out, a problem has been analysed, a decision has been reached, critical information has been exchanged, which would trigger the need for some information to be written down. Thus, the analytic evaluation has shown that the Meeting Miner can *"reliably"* locate information among a comprehensive amount of these *significant* meeting events.

Finally, we would argue that the main advantage of performing an analytic evaluation is that it essentially removes the subjectivity which inevitably comes with experiments involving participants. On the other hand, the analytic evaluation limitations lies in the very fact that it does not account for the preferences and interaction styles of human subjects, who might often find the system better or worse than one would have expected, often for reasons one could not have suspected and who will almost inevitably use the system in a way one could not have anticipated. This is the subject of the usability studies and task-oriented information retrieval task described in the following chapter.

Chapter 9

Meeting Miner Heuristic, Usability & Task-Oriented Evaluation

9.1 Introduction

The analytical evaluation of the Meeting Miner described in the previous chapter was useful to provide quantitative measurements of the system's novel interaction-based search functionalities. Precision & recall showed that the Meeting Miner's content-based retrieval system can be used reliably with time-based techniques alone or with time-based and ASR techniques combined.

In this chapter, we are primarily interested in evaluating the Meeting Miner from a user's perspective, with our main concern being two-fold (i) how practical end-users find the browsing system and (ii) how well do they perform? These are very difficult questions to answer, chiefly because these concepts can be quite subjective. In order to conduct an evaluation of the Meeting Miner as thorough as possible, we have selected and performed several types of evaluations, each of which translated in improvements to the system. In this chapter, we describe the motivations, methodology and results of four separate evaluation: a heuristic evaluation, a pilot usability study involving 10 users and finally, a task-oriented evaluation involving 20 participants, which was coupled with our second and last usability study.

In what follows we first present a quick overview of interactive systems evaluation methods. We then formulate a number of questions one might wish to answer while investigating the performance of the Meeting Miner browsing tool. This is followed by explanations behind the choice of our evaluation methods, a detailed description of the evaluation methodologies selected and finally a description of the evaluation experiments and their results.

9.1.1 Overview of Interactive Systems Evaluation Methods

A number of tried and tested methodologies have been developed over the years as benchmarks for interactive systems and user interface evaluation. Among these are: *heuristic evaluation*, *predictive evaluation*, *software guidelines*, *cognitive walkthrough*, *usability testing* and *field studies* (Jeffries et al. 1991, Preece et al. 2002). In a heuristic evaluation, evaluators forms an opinion of the user interface by comparing it to a set of criteria (heuristics) which are known to lead to potential usability problems. In a predictive evaluation, system designers rely on user-models or their knowledge and experience of typical users' behaviour to anticipate and rectify potential usability issues. In a cognitive walkthrough, evaluators explore an interface by performing the core tasks (e.g. sequence of actions) expected to be carried out by typical users of the system. Usability testing will generally involve a set of experiments involving a number of users in a "laboratory" environment: the performance of the system is then measured according to certain criteria (completion time, score, questionnaire). Finally, in a field study, the system is observed in it's natural environment (e.g. observing the usage of a booking software in a travel agency).

9.1.2 Goals of the evaluation

Before evaluating an interactive system and in order to device appropriate experiments and evaluation methods, one first needs to consider the type of information one is seeking to learn from the prototype and its' evaluation. List 9.1 illustrates some of the potential questions one will be hoping to answer after having carried out an evaluation of the Meeting Miner.

List 9.1 Goals of Meeting Miner evaluation

- *to what extent does the Meeting Miner facilitate browsing of meeting recordings?*
- *to what extent does the Meeting Miner help users construct a structured mental image of a meeting recording?*
- *to what extent does the Meeting Miner facilitate identifying information of interest?*
- *to what extent does the Meeting Miner facilitate discovering information of interest?*
- *to what extent does the usage of the Meeting Miner browsing tool result in increased performance in a task-oriented experiment?*
- *which of the Meeting Miner browsing functionalities are the most useful and why?*
- *which of the Meeting Miner browsing functionalities are the least useful and why?*
- *do users find the functionalities of the Meeting Miner easy to learn and use?*
- *what could improve the Meeting Miner functionalities or user interface?*

The above list highlights the fact that a single evaluation method or experiment is very unlikely to be sufficient on its own to answer all the usability and performance questions relating to the Meeting Miner browsing tool. As a result, we decided to perform: (i) a heuristic evaluation in order to uncover potential usability issues and to improve functionalities and/or the user interface, (ii) a pilot usability study in order to get some users' feed-back on the browser and find out which features are considered most useful and which are considered less useful, and finally (iii) an information retrieval task, in order to compare user performance on the Meeting Miner compared to a Random Access Interface (RAI), coupled with (iv) a second and final usability study. A summary of the evaluation methods performed on the Meeting Miner, including the analytic evaluation described in the previous chapter, is detailed in Table 9.1. In total and including the analytic evaluation performed in the previous chapter, there was 5 evaluations performed on the Meeting Miner, involving 36 distinct subjects.

Table 9.1: Evaluation methods

Evaluation Method	Number of participants	Nature of Experiment	Measurement Method
Analytic Evaluation	-	random samples of terms from meeting evaluation subset	Precision Recall
Heuristic Evaluation	6	4 x (1 hour) usability sessions	Report
Pilot Usability Study	10	Demonstration + 1 hour evaluation session	Usability Questionnaire
IR task-oriented evaluation	20	2 x 25 min. experiments using 2 different interfaces	Task Performance
Final Usability Study	20	as above	Log of actions + Usability Questionnaire

Remark on Notation

Throughout this chapter, comments from users will be emphasised and in quotation marks to distinguish them from the author's comments. This is an example: *"user's comment"*.

9.2 Heuristic Evaluation

Heuristic evaluation of user interfaces was first formalised by the work of Nielsen and Molich (1990). They proposed an evaluation method which involves a (small) number of evaluators formulating opinions of a user interface according to a (small) number of rules of thumb (heuristics). These heuristics are developed over the years through experience of interaction systems and their users, mainly because they are known to relate to potential usability problems in the system. Examples of such heuristics can be issues relating to status' visibility, consistency, feedback, control, etc. The

main advantages of the heuristic evaluation are that a small set of rules, combined with a small number of evaluators are sufficient to uncover a majority of all potential problems and that the evaluators need not necessary be experts (even though these are obviously better at uncovering potential problems). Interestingly, individual evaluators will typically find only a small proportion (e.g. less than half) of potential usability issues. However, this is compensated for by the fact that aggregated results of a small number of evaluators are sufficient to find a majority of usability issues. Nielsen and Molich (1990) found that, on average, sets of ten evaluators were able to find between 70% to 97% of usability problems in four usability experiments. It is also not always necessary to have a running system before performing the evaluation, as users can anticipate a number of usability issues from specification and descriptions of a prototype alone. The heuristic evaluation can thus be performed early in the system's life-cycle and save the system's designers from many potential pitfalls at later stages of the design process.

9.2.1 Methodology

The heuristic evaluation of the Meeting Miner was performed on a running prototype. Six computer science students in third year Information and Communication Technology (ICT) at Trinity College took part in the evaluation. These students followed a course in Human-Computer-Interaction (HCI) for a full term (12 lectures), during which they were introduced to principles of interactive system design, using the textbook by Preece et al. (2002). In addition to these lectures, the students coursework consisted of an evaluation project of the Meeting Miner which took place over 4 one-hour tutorials. The first tutorial consisted in an overview of multimedia indexing and navigation techniques and an introduction to existing meeting browsing tools (Bouamrane and Luz 2007b). In the following two tutorials, a demonstration of the basic functionalities of the Meeting Miner was performed after which the students explored and used the browser as they pleased. They were also asked to answer a number of questions about a specific meeting and this experiment was later used as the basis for the usability study detailed in section 9.3. In the final tutorial, the students were encouraged to discuss the potential flaws of the system using a Heuristic Evaluation Toolkit available through the Internet resources of Preece et al. (2002)'s textbook. Finally, the set of tutorials culminated in the handing in of heuristic evaluation reports (Appendix D).

9.2.2 Results of the Heuristic Evaluation

Visibility of Contributions

One evaluator suggested that the contributions of individual participants in the meeting outcome was not obvious and that highlighting parts of the text according to authors could enhance a user's understanding of the meeting.

Visibility of Keywords

One evaluator complained that only certain keywords were highlighted in the Indexer panel and that it was not possible to select specific keywords and higlight them.

Information Mapping and Consistency

Several evaluators suggested that the Audio View Panel (at the top of the user interface) be coupled with the Audio Control Panel (at the bottom of the interface).

Visibility of Information

Some evaluators thought that the space allocated to the meeting outcome was too large as the main source of information during meeting browsing comes from the audio recording. They also thought that seeing participants' actions (Participants' Insertion panel) in real-time was more important to understand the meeting than the final outcome itself and therefore that this panel should be of bigger size.

Tailorability

As a result of the previous remark, some users suggested to allow for the ability to tailor the user interface to one's needs, i.e.: to be able to display only features or functionalities which one is currently using. The evaluator suggested removing or adding the participants' panel, keyword and topic search panels when judged appropriate by the user. As an example, he said that the participants insertion panel and document panel could be somehow redundant.

Navigation Feedback

Some evaluators highlighted the fact that consecutive searches could include segments of speech which had already been listened to, leading to some frustration. They suggested to introduce an option to skip segments of the speech recording which had previously been listened to. Another issue highlighted was that, as the keyword search skip to arbitrary locations in the meeting, it was sometime hard for the user to know which part of the meeting he was currently listening to. In particular, it was sometimes hard to know if the current audio segment being listened to had already been heard (as certain speech segments can occasionally show a lot of similarity) or even if it was close to a location in the recording which had already been listened to.

Controlled Vocabulary too constraining

Several evaluators found the use of a controlled vocabulary too constraining and suggested that the system should allow for any words to be searched.

Ambiguity of Complex Query - Topic Search

Although participants clearly understood that a keyword search looked for instances of a specific term in the recording, the complex query search was more ambiguous and the evaluators found it unclear if the system returned speech segments containing the words in the query when found separately (OR-search) or together (AND-search).

Provide Visual Summary and Topic Segmentation

One evaluator suggested that the system provide a short visual summary of the meeting where one could see at a glance the main topics covered during the meeting.

Provide additional meeting information

One evaluator suggested providing information and access to the additional resources which were used during the meetings, such as text documents and graphs, websites visited etc.

Visibility of Participants

One evaluator suggested using photos of participants instead of the avatars in order to give a more "human feel" to browsing the meeting, which he suggested would make it easier to follow the meeting.

Adding Video

Several evaluators suggested to add a video panel so participants could see each other, gesture to each other and see each others' reactions.

9.2.3 Meeting Miner functionalities rated satisfactorily

Overall, the users found that the Meeting Miner fulfilled its' purpose as a meeting browsing tool reasonably well. The general feeling was that it permitted finding specific information about a meeting in a shorter amount of time that would be needed by random access only. Navigation was judged satisfactory, with the keyword search found to be the most useful in finding specific information. The time slider in the Audio Control Panel gave an indication of location within the meeting recording at all times, while the Audio View Panel permitted finer grain local navigation. Feedback was judged good, with the audio control button graphically displaying at all times the status of audio play-back (playing, stopped, past or further relevant speech segments, etc.) Participants' edits displayed in the Participants' Action Panel and graphical display of speech activity (silent, talking) was also judged positively as the users did not know the meeting participants personally and relied on the highlighting of participants' avatars to distinguish between participants during crosstalk. The functionalities were clear and easy to use and the use of unnecessary window

pop-ups had been avoided. The mouse was the only device required to use the system and this was considered an advantage by most of the evaluators. The system was found to be flexible and responded to most navigation needs. The paragraph-based browsing was found too complex by many however and did not correspond to some intuitive mental image of the meeting recording. As a result, it's use was limited to some initial experimentation, after which most users reverted to using the more familiar linear Audio View (random access). One user did however found it quite useful and used it for coarse-grained navigation followed by finer-grained local navigation using the Audio View. This corresponded to the sort of navigation behaviour we had initially envisaged for the paragraph-based browsing modality.

9.2.4 Recommendations which were implemented (or not) in the Meeting Miner

Visibility of Contributions

An option was introduced in the menu which highlights the meeting outcome according to participants' contributions (see Figure in implementation chapter 7.20).

Visibility of Keywords

The Meeting Miner initially only displayed index terms with highest term frequency on the Audio View. However, it is a well known fact of information retrieval that the most common words are not necessarily the most informative or discriminative (van Rijsbergen 1979) (this is why metrics such as TF-IDF are commonly used instead). In the Keyword Panel, we added an option to chose to display index terms either according to term frequency or inverse term frequency. In addition, any keyword selected and added to the Topic Panel would be individually highlighted. These new functionalities are illustrated in Figure 9.1.

Visibility of Information & Tailorability

As the Meeting Miner User Interface was designed using the Java Swing Split Pane component, the user can already adjust to some extent the size of individual panels. We thought that for the purpose of evaluating the Meeting Miner, this was a sufficient level of UI displaying flexibility and we did not implement further tailorability of the interface as suggested by some of the evaluators.

Navigation Feedback

This is one of the most important usability issues highlighted by the evaluators. Having to listen several times to the same speech segments, sometimes without knowing for sure if this particular segment had already been listened to, was indeed susceptible of causing great frustration to the

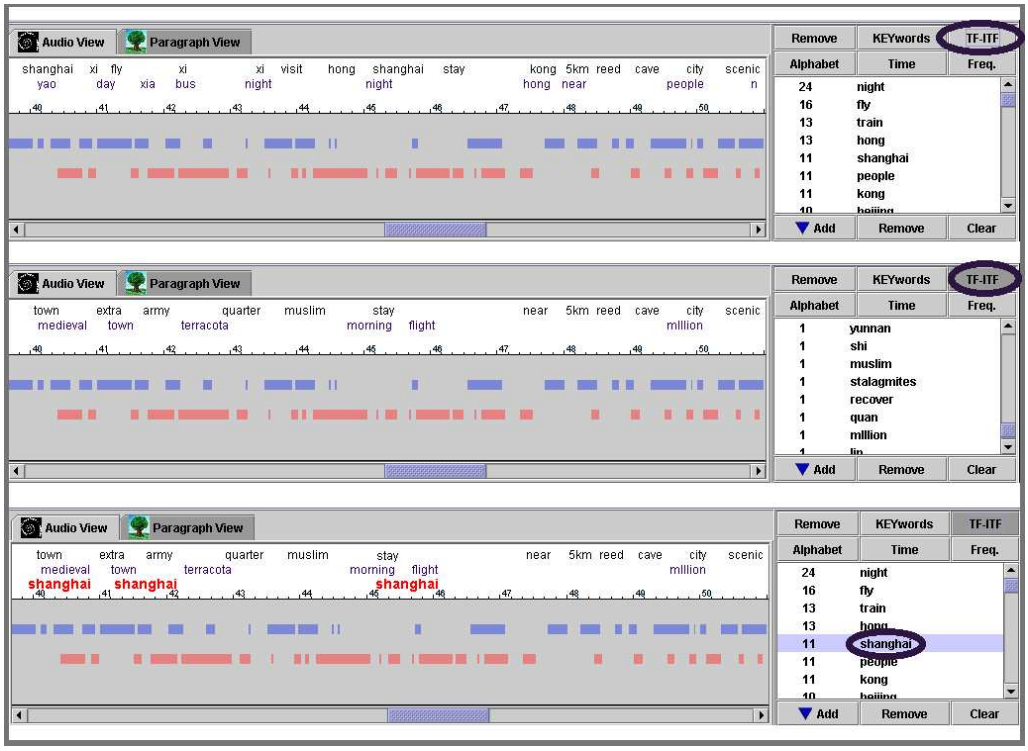


Figure 9.1: Three modes of displaying index terms on the audio view: (i) Term Frequency, (ii) Inverse Term Frequency or (iii) selecting a specific term

potential users of the system. As a result, several steps were taken to give greater audio navigation choices to the users and the following listening options were added to the system:

Skip Listened: this option will skip any audio segment which has already been played

Skip Silence: this option will skip all silences

Cut Audio: with this option, the user can select specific speech segments which will not be played back. The Cut selection can be undone, and the user can also still listen to a Cut audio segment, should he decide to, by clicking on it. Finally, all Cut selections can also be cleared

Audio "Wear Marks"

In addition, audio "wear marks" were introduced in order to graphically display which parts of the audio recording had already been listened to. A similar idea was first described by Hill et al. (1992), who implemented two text applications called Edit Wear and Read Wear, which graphically display the location and amount of editing and reading activity on documents. To the best of our knowledge, we do not know of the idea of "wear marks" having been applied to audio or speech interfaces. These audio wear marks, along all the new audio navigation options introduced in the Meeting Miner as the result of the heuristic evaluation, are illustrated in Figure 9.2.

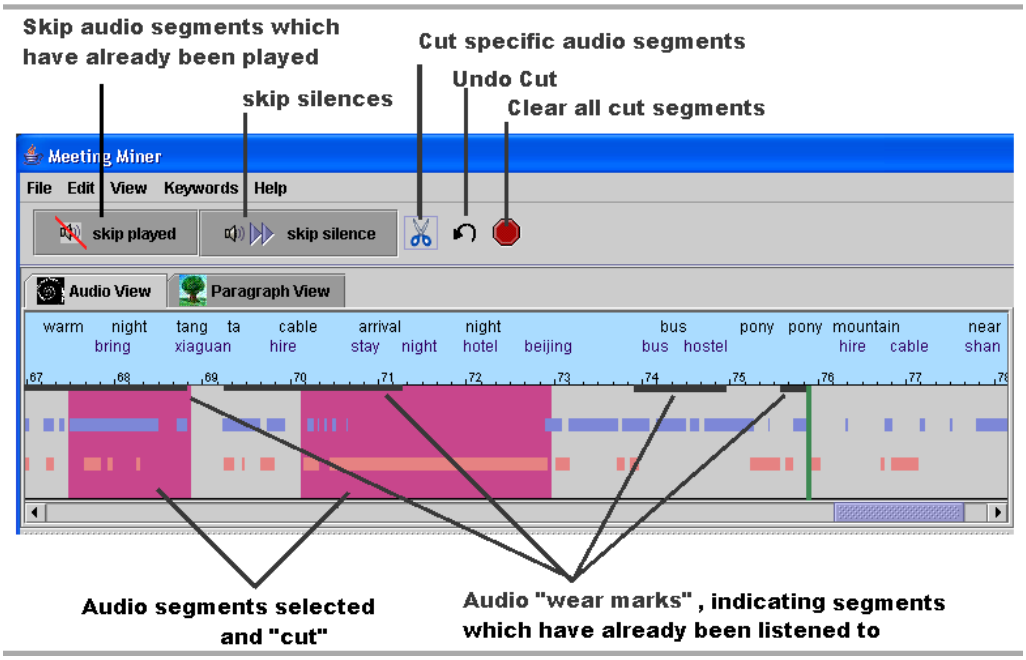


Figure 9.2: Greater audio navigation options introduced as a result of the heuristic evaluation

Ambiguity of Complex Query - Topic Search

This was also found to be a very important usability issue: if users are to browse a meeting recording efficiently, they need to have a clear mental image of what the system and its' functionalities do and thus, it is preferable if their understanding of the system is as close as possible to the actual underlying search mechanisms. The initial complex query search did an AND-search by default: it only returned speech segments if the temporal distance between terms of a query was less than a certain time threshold τ. However, there were many occasions when users when interested in listening to one *or* the other terms of a query, without having to perform several keyword searches. As result, the Meeting Miner's complex query search was modified to provide a greater degree of flexibility, by including two search modes: AND & OR as described in 7.3.4. The user interface was also modified to clearly display the status of the current search mode, which is illustrated in Figure 9.3.

Figure 9.3: Flexibility of Topic Search: AND & OR mode, which are explicitly displayed in the user interface

Other usability issues

Of the remaining usability issues, we distinguish two types: ones which are related to the browsing modalities of the Meeting Miner and others which are related to the nature of the multimedia recordings. Examples of issues relating to the nature of the multimedia recordings are suggestions to include video streams or other resources accessed during the meetings. These are not usability issues relating to the Meeting Miner, as the corresponding data is simply not available. Including video in the browser would first require introducing video streams in the meeting capture environment.

On the other hand, an example of usability issues directly related to the browsing modalities of the Meeting Miner was the decision to implement a controlled vocabulary. This is definitely a restriction we imposed on the system in order to achieve reasonable results. As evidenced by results from the pilot usability study described in the next section, users will ignore certain functionalities (e.g. ASR speech transcripts) if these fail on several occasions and are thus deemed "unreliable". Finally, while we would totally agree with the remark made by the evaluator who suggested providing a visual summary with the main topics discussed during the meeting, unfortunately this is not a trivial problem and was not implemented in the Meeting Miner.

9.2.5 Concluding Remarks on Heuristic Evaluation

As previously mentioned, the main advantages of the heuristic evaluation are that: (i) it only requires a small amount of evaluators in order to uncover a large amount of usability issues, (ii) evaluators need not necessarily be experts and in our case were computer science students who had recently been introduced to HCI design, (iii) it can be performed early in the system's life cycle. The most important issues regarding the Meeting Miner highlighted by the evaluators concerned the ambiguity of the topic search, limitations of the keyword indexing on the audio view, and finally, limitations in the audio navigation. Once these issues were addressed, we were then ready to run a larger pilot usability study of the Meeting Miner, described in the following section.

9.3 Pilot Usability Study

The goal of the first usability study was to test the new Meeting Miner system including the latest modifications performed as a result of the heuristic evaluation described in the previous section, to ensure that the latest modifications did not introduce new usability issues, to get some feed-back from users by getting them to fill in a usability questionnaire and finally, to prepare the ground for the information retrieval task described in the next section.

9.3.1 Methodology

This study involved 10 final year computer science students, who were not involved in the previous heuristic evaluation. Each session began with a demonstration of the software, which lasted around, 10 to 15 minutes. The participants were then encouraged to explore the functionality of the Meeting Miner and were free to ask any questions about the system. This familiarisation step with the prototype generally lasted 15 minutes to half an hour. Once the participants were confident they could use the system reasonably well, a meeting was loaded and they were asked to answer a number of questions about the meeting in a set amount of time. At the end of the experiment, they were asked to fill in a usability questionnaire. The whole experiment lasted around one hour to one hour and a half per participant.

Usability Questionnaire

The usability questionnaire, a sample of which can be seen in Appendix E, is divided into three parts: questions about the task itself, questions specifically relating to audio aspects of the experiment, and finally questions relating to the Meeting Miner browsing tool. When appropriate, users were invited to assign a score on a scale from zero to ten to certain features of the experiment or the prototype. In addition, the participants were invited to make precise comments about the questions whenever they deemed it necessary.

9.3.2 Results of Usability Evaluation

Users were ask to rank the most useful graphical components and functionalities of the Meeting Miner. Their preferences are illustrated in Figure 9.4. The functionalities judged the most useful are essentially the ones relating to keyword indexing and keyword search. The next most useful set of functionalities are the ones relating to speech indexing and navigation. Finally, the speech transcript panel and the paragraph-view panel were judged the least useful. Participants' first preference choices are summarised in Figure 9.5.

Here are some excerpts of comments participants wrote in the usability questionnaire which explain their first preference choices and their like and dislike of certain functionalities of the Meeting Miner.

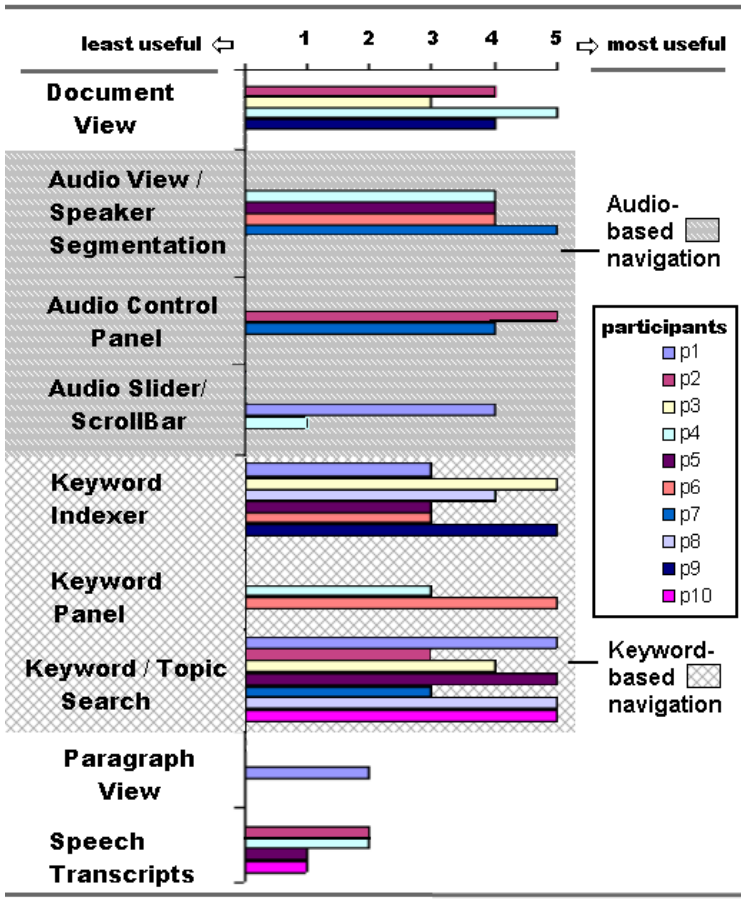


Figure 9.4: Participants' ranking (according to usefulness) of components and functionalities of the Meeting Miner

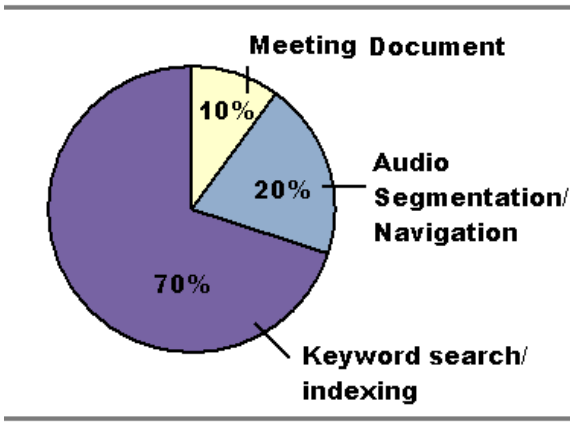


Figure 9.5: Participants' first preference choice in the Meeting Miner functionalities

---

Question 9.1 How useful was the ability to search for keywords?

- *"Very useful, all my answers were based on it"*
- *"it was a means to navigate the conversation"*
- *"does exactly what it's meant to .. and is the most useful thing to do!"*
- *"I think this element was crucial"*
- *"makes backtracking through audio easier"*
- *"it would be missed if it wasn't there"*
- *"useful to do the task (questions) but sometimes didn't work"*
- *"skips the tripe"*

Question 9.2 How useful was term indexing above the audio view?

- *"it lets you skip unimportant bits"*
- *"made searching much easier"*
- *"enabled me to pinpoint keywords and topics"*
- *"allows you to skip from section to section while staying on topic"*
- *"helped narrow search"*
- *"allowed you to skip ahead"*

The following answers to Question 9.3 sheds some light on the reasons behind the Meeting Miner least popular components: the ASR transcript Panel and the Paragraph View. Users seem not to have found these functionalities useful for fundamentally different reasons: for the ASR transcripts, the main reason was that, with a average WER of 60% , the quality of the transcripts were just too poor to be considered of any use. An interesting lesson from the usability study was that if users find that a feature of the browser is not performing well on several occasions, they will quickly give up on it and just ignore it. As a result, we subsequently removed the ASR transcript panel from the Meeting Miner (although the systems still uses ASR transcripts for underlying keyword and topic search as described in the previous chapter).

Question 9.3 Which components/ functionalities did you find not useful or didn't use and why?

- *"(speech) transcription, didn't understand the sentences... was inefficient"*
- *"(speech transcripts) didn't make sense"*
- *"speech transcripts: it was easier to recognise information from speech"*
- *"speech recognition: supposed transcript of recording , mostly incoherent ... seemed wholly inaccurate"*
- *"speech (transcripts): never used it"*
- *"paragraph (view) was not useful as it seems too complex"*
- *"I found it all useful but perhaps I did not use the paragraph view as much..."*
- *"didn't use paragraph view. Quite confusing to use, wasn't very clear how it worked"*

The main reasons why users did not find the paragraph-based search useful was because they (i) did not really understood how it worked and (ii) found it too complex to use. However, a small number of users did use it for coarse-grained search, in combination with local audio navigation for finer-grained search. This would seem to indicate that in order to be considered useful by the novice users, browsing functionalities needed to be clearly understood and match some more familiar navigation concepts (i.e. random access).

Finally, Question 9.4 describes possible improvements to the Meeting Miner suggested by the participants of the study. The dominant suggestion is to improve the keyword search functionality by expanding the search to any queries, while improving the "precision" of the search. In other words, it would seem that many users found that performing a search using a controlled vocabulary, the very reason of which was to improve the Precision of search in the light of prohibitively high word error rates in meeting transcripts, was a serious limitation to the system.

Question 9.4 What do you think could improve this system?

- *"audio should stop after the end of a section related to a keyword"*
- *"allowing the user to search for his own keywords"*
- *"more defined keyword search where you can type in the desired word"*
- *"more accurate speech recognition"*
- *"search for phrase, word wasn't enough sometimes"*
- *"more specific keyword for searching"*

9.3.3 Concluding Remarks on Usability Evaluation

The pilot usability evaluation fulfilled a number of goals: (i) it made sure that all major bugs had been eliminated from the system (ii) it made sure that there was no major usability issues which could impede the use of the system (iii) it gave us a better understanding of how subjects used the browsing tool in a practical task, giving us some precious insight in their speech browsing behaviour (iv) it was an opportunity to collect users' feed-back through the usability questionnaire and thus to understand which components were deemed useful or not and why (v) finally, it was an opportunity to test the system in a task-oriented environment and thus prepared the ground for the next step of the Meeting Miner evaluation: the information retrieval task described in the following section.

9.4 Browser Evaluation Test - Information Retrieval Task

9.4.1 Goals of the Information Retrieval Task

While the heuristic evaluation and usability study highlighted, on one hand, some of the usability issues, and on the other hand, some of the strength of the Meeting Miner, they fail to give us some insight as to *if* and *how much* the Meeting Miner potentially improves a user's meeting browsing experience and performance. In order to evaluate users' performance with a practical task, we designed a task-oriented information retrieval experiment. Recently, task oriented evaluations of IR systems has started to take a prominent role as many researchers in the field have realised that involving users' in the evaluation loop is equally, if not more important, than solely relying on purely analytic methods (Hersh et al. 1996, McLellan et al. 2001).

However, the first question one needs to answer before going about designing an information retrieval experiment is what exactly are the goals of this experiment? The answer to this question is: *to provide some measurement of the information retrieval system's performance.* The answer to the first question leads yet to more questions: *what measurement and what performance?* A reasonable answer to the latter is that evaluating the system performance consists in offering some indication at how good the system is at supporting users in the tasks they are expected to perform while using the system. Which brings us to the next set of questions: *which are the tasks users would be expected to perform using the system?* Since the system was designed as a meeting browsing tool, essential functionalities which need to be supported by the system are: *efficient navigation* and identification of *precise information* within recordings in a *minimum amount of time.* As an analytic evaluation of the search features has been performed in Chapter 8 and a qualitative evaluation of the system through the heuristic and usability evaluation of the Meeting Miner has been described in the previous sections, we are here interested in gathering quantitative measurements of the system's performance.

9.4.2 Methodology

The Browser Evaluation Test - Motivation

We decided to design an information retrieval tasks along the lines of the BET (Browser Evaluation Test) proposed by Wellner et al. (2005). The main motivation behind the BET was to provide means of comparing meeting browsers' performance and browsing functionalities regardless of meeting corpora and browsing modalities. Until then, the lack of common metrics or publicly available evaluation corpus meant that evaluations on meeting browsers were usually performed by the browsers' designers using some subjective evaluation method, as described in chapter 2.5. In order to address the issue of the lack of publicly available meeting corpus, the AMI (Augmented Multiparty Interaction) project (*AMI Project* 2006) has recently released a multi-modal meeting corpus consisting of a 100 hours of co-located meeting recordings, involving a small number of people in a variety of scenarios (Carletta et al. 2005, AMI Corpus 2006).

BET - Methodology

The BET is designed as follows: using the system to navigate a meeting recording, a user needs to answer as many questions as he can in a certain amount of time. The questions are related to a number of *"observations of interest"* gathered by independent observers. The BET methodology comprises four main components: (i) a meeting corpus, (ii) observations of interests, (iii) an evaluation test, and finally some iv) performance metrics. The meeting corpus is simply a collection of any type of meeting recordings. Observations of interest are produced by independent *observers*, who can watch an evaluation meeting for as long as they wish and take notes of interesting events. Observers should have no interest in the development of the meeting browsing system to avoid experimenters' bias. The methods suggest several observers, as the most interesting meeting events are likely to be spotted multiple times (thus, truly validating these as observations of interests). The evaluation test then consists of presenting True-False statements about these observation of interest to test subjects, who must pick the correct statement. Test subjects answer the questions one at a time and the test duration is half the duration of the meeting recording. Answers are timestamped with time of answer and media time (in meeting recording). Finally, a BET score consists of a pair of metrics: (i) speed (answers/minute) and (ii) precision (percentage of correct answers). The BET methodology has previously been used to evaluate a number of browsing systems, including the Ferret media browser (Wellner et al. 2004) in the experiments described in Wellner et al. (2005).

BET Limitations

There are several limitations to the BET in it's present form. First of all, it does not resolve issues relating to diversity of meeting corpora and meeting browsing modalities so in order to really

permit systems comparisons, meeting browsers should be evaluated on the same meeting corpus. Standard meeting corpora, such as the AMI meeting corpus (Carletta et al. 2005, AMI Corpus 2006) have become available which might help alleviate this problem. However, even this might not be sufficient to resolve comparison issues on it's own, as certain meeting browsing systems, such as the Meeting Miner, rely on meeting metadata produced during meeting capture (Geyer et al. 2003, Bouamrane and Luz 2006a) which are not necessarily available in standard meeting corpora. The fact is that no single evaluation test can possibly solve issues relating to the diversity of meeting corpora. Meeting is such a rich and diverse activity that researchers will always be interested in analysing particular aspects of meetings, whether collocated in purpose-built meeting rooms (Lee et al. 2004) or in Internet-based environments (Erol and Li 2005).

What we perceive as being the real limitation of the BET is the limited number of proposed metrics: speed and precision (also referred to as *accuracy* in the original methodology). The metrics are coarse-grained and hard to interpret. They give little information about the evaluation task or subjects' browsing behaviour. The BET score is insufficient in pointing to strength or flaws of a system, and as a result, is of limited use should one wish to use the evaluation in order to improve a browsing system.

In what follows, we describe how (and highlight when) we adapted the BET recommendations to evaluate the Meeting Miner using meeting recordings from of our corpus. We also describe our proposal of a number of novel performance metrics to complement the BET evaluation framework. The aim of these additional metrics is to provide quantitative measurements of how successfully a meeting browsing system supports users during an information retrieval task. Measurements include how quickly the browsing system enables to identify the first useful information, give an indication of how broadly a meeting recording has been accessed, and finally, a combination of some of these metrics is used to provide an overall user browsing score for a specific IR task. In addition, the metrics are designed to they can easily be interpreted in order to provide some insight into the meeting browsing system's usage. Finally, they remain general enough so to be applied to any type of browsing system.

Experiment Design

The information retrieval experiment is designed as follows: participants listen to two different meetings for 25 minutes each and answer as many questions about the meeting as they can, using a different interface for each meeting: the Meeting Miner in one task and a Random Access Interface (RAI) in the other (BET Base condition). Participants were first introduced to the two interfaces using a demonstration meeting recording which was not used in the following evaluation experiment. Once the participants felt confident in using the interfaces, the experiment proceeded to the actual evaluation task. This familiarisation step usually lasted between 15 to 45 minutes. Participants had a break of 5 to 10 minutes between the two experiments, in order to ensure they

remained focused on each task. At the end of the experiment, participants were asked to fill a usability questionnaire similar to the one of the usability study described in the previous section. This was followed by an informal discussion in which participants were encourage to express their views or comments about the software and the task. In total, each experiment lasted about an hour and-a-half.

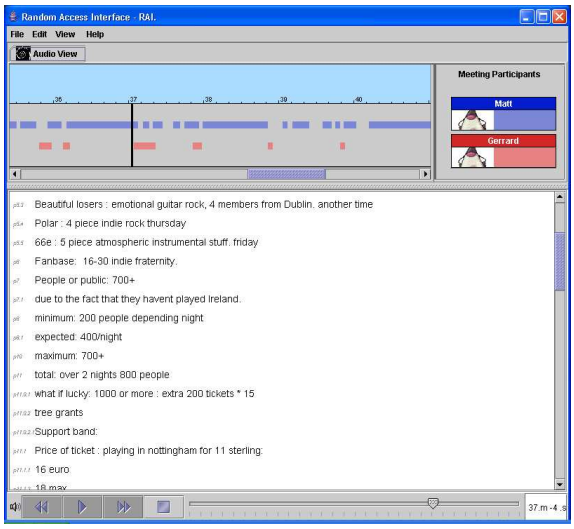


Figure 9.6: The Random Access Interface

The RAI system can be seen in Figure 9.6: it displays the meeting document outcome and offers the same random access to the speech recordings as the Meeting Miner. The main differences between the two interfaces is that the RAI has none of the keyword indexing and search functionalities of the Meeting Miner and it does not display contextual information associated with paragraphs (paragraph view). A similar experimental setting was used in the evaluation of the SCAN speech messages retrieval system (Whittaker et al. 1999). The performance of the SCAN system, including ASR transcripts and keyword indexing, was compared against a "tape recorder" interface which only permitted selecting and listening to messages without any additional information or functionalities.

Meeting Questionnaire

The meeting questions used in the IR task were first devised independently by three individuals: the author and two final year computer science students whose final year projects involved working on the meeting corpus collection and speech processing. The only criteria for choosing a question was that the answers could not be easily guessed (BET recommendation) or inferred from reading the meeting outcome alone, so that access to relevant sections of the speech recording was necessary. The three individual sets of questionnaires produced for both meetings were subsequently merged into a single set of questions for each meeting. Preference was given to questions which appeared in all three sets, while questions which were deemed too easy or irrelevant were discarded. As

suggested by Wellner et al. (2005) the assumption here was that the most interesting aspects of a meeting are generally noticed multiple times by several observers. The final questionnaire consisted of 15 questions for each meeting, which was designed to be more than any participants could answer within the (25 minutes) experiment duration (BET recommendation). The details of these questions, as well as the corresponding answers, can be found in Appendix F.

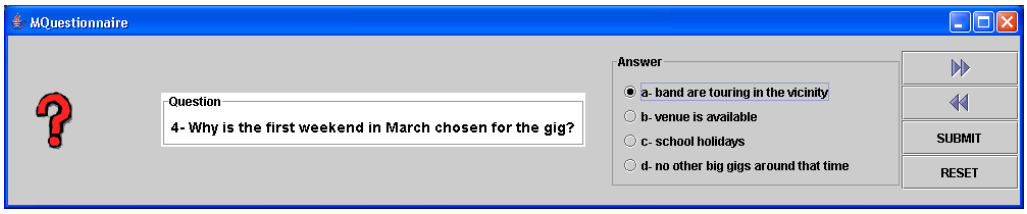


Figure 9.7: The Meeting Questionnaire User Interface

The Questionnaire User Interface (QUI) can be seen in Figure 9.7. Only one question is displayed at a time, along with 4 multiple choice answers. Each time the questions are first loaded by the interface, their order of appearance is shuffled (BET recommendation) so as to remove the possible bias in the ordering of easy or difficult questions. Users however are free to skip questions at any time, and as these are ordered in a round-robin way, they will jump back to the first question once they reach the last one. Users can also navigate back-and-forth between the questions. However, once a question has been submitted, it disappears from the list and the participant can not modify his answer. Answers are timestamped and the answering order is recorded by the QUI. During the experiment, the questionnaire interface is displayed along the browser interface (Meeting Miner or RAI) as illustrated in Figure 9.8. The experiment duration or the remaining time were not displayed on the QUI. This was a deliberate choice as we wanted to avoid the phenomenon of last-minute guesses, as observed by Wellner et al. (2005), whereby participants answer as many questions as they can in the final moments of the experiment. Instead, time was kept by an observer (the author) and participants were allowed to enquire about the remaining time whenever they wished. They were notified of the end of the experiment when the evaluation time elapsed.

Choice of Evaluation Meetings

As each experiment was set to 25 minutes, the evaluation meetings needed to be longer (BET recommendation), otherwise, sequential listening of the recording would have been sufficient to answer most of the questions, thus defeating the purpose of evaluating the browser. This criterion alone discarded a significant number of the shorter meetings for the IR task. Another criterion was that the topics of the meetings should be understandable by anyone (BET recommendation), which excluded some of the more technical recordings of our corpus, such as the ones which dealt with meeting browsing technology, ASR etc. The two meetings eventually selected for the

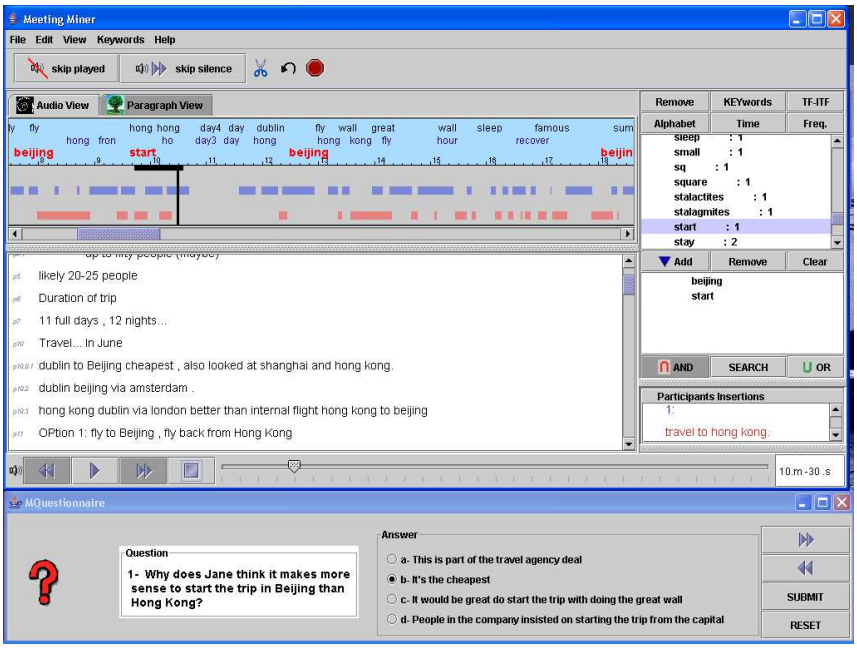


Figure 9.8: The IR task experiment: Meeting Miner and Questionnaire User Interface

evaluation tasks were meetings A and B (see evaluation subset Table 8.1 in previous chapter). Meeting A deals with the organisation of a concert in Dublin and involved selecting bands, a venue, advertisement and other organisation issues. In this meeting, which lasts slightly less than an hour, participants would have been familiar with names and places. Meeting B, which lasts nearly an hour and-a-half, deals with the organisation of an 11 days itinerary of China, this was considered by many participants as a much harder meeting, as many names and places would have been very unfamiliar to most users. In order to avoid a possible bias of the systems toward a specific meeting, the combination of meetings and interfaces was swapped for every second experiment as described in Table 9.2. Finally, in order to remove the possible bias of using one interface or one meeting before another, the use of the meetings and interfaces were also switched around every second experiment (BET recommendation to avoid any sequence effects).

Table 9.2: Interfaces used for IR evaluation according to each experimental condition

Experimental Conditions	Meeting A	Meeting B
Participants 1-10	Meeting Miner	RAI
Participants 11-20	RAI	Meeting Miner

9.4.3 Evaluation Metrics

Once the evaluation experiments have been conducted, one needs to use a number of metrics to measure users' and systems' performance. We use and adapt some of the following measurements suggested by the BET as indication of the system's performance:

Recall - (ρ): How many questions have been answered correctly in a set amount of time? In our case, the score is defined as a percentage of the total number of questions. The assumption here, is that although users will probably not have the time to find all the necessary information in order to answer all the questions, they are able to scan through all the questions using the round-robin QUI.

Precision (π): (also referred to as *accuracy* in the original methodology.) Of all the questions answered, how many of them have been answered correctly? So we calculate precision as the ratio of correct answers over the number of questions answered. This is an indication of how good the system is at identifying the correct information or if, on the contrary, it is misleading.

Average Answering time (Avg. A.t.): it takes to answer a question over the duration of the experiment?

In addition, we propose the following metrics to further measure meeting browsers' performance:

First Correct Answer (FCA): The First Correct Answer-time (FCA-t) metric corresponds to the time it takes for participants to submit their *first correct answer* during the experiment. The reason why we believe this to be a significant metric is that it takes into account and counter-balances the phenomenon of information accumulation as observed by Wellner et al. (2005): as the experiment goes on, participants get better at answering questions because they have become more familiar with the meeting content or the questions of the experiment. In some cases, they might even have already stumbled across the relevant information while previously searching for a separate piece of information. The longer the duration of the IR task, the least obvious are the benefits of using a browser over sequential play-back of a meeting. Thus, the FCA gives an indication of how quickly the system permits users to locate and identify the *first* piece of useful information (observation of interest). The concepts of information accumulation and FCA-t are illustrated in Figure 9.9, where the participant only manages to answer one question in the first ten minutes of the experiment yet managed to answer a further seven during the last 15 minutes. An alternative to measuring FCA in terms of time would be to calculate it as a percentage of the experiment duration (or meeting duration) : FCA-p = $\frac{FCA-t}{experiment\ duration}$. Using the example in the figure, we would find a FCA-t = 5 min. 30 s. and FCA-p = $\frac{5.5}{25} = 22\%$.

Information Span (IS) and Information Accessed Ratio (IAR): We define a meeting recording Information Span (IS) as the time span of *critical information* a user needs to access in order to complete a specific task. In a real world scenario, this corresponds to the span of all the *critical information* one needs to access in order to fulfil one's information needs. Using the BET's concept of *observations of interests*, the information span would correspond

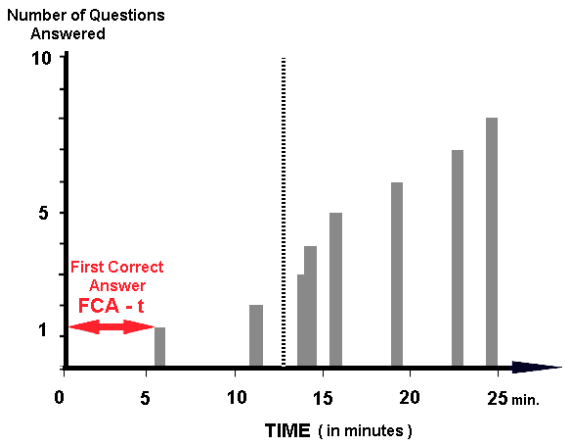


Figure 9.9: Information Accumulation and First Correct Answer-Time (FCA-t)

to the media time-distance between the first and the last of these observations of interest necessary to complete the BET. In our information retrieval task, a meeting IS corresponds to the time-distance between the critical information necessary to answer the first and the last questions. The concept is illustrated in Figure 9.10 which shows which parts of the evaluation meeting A a participants needs to listen to in order to answer all 15 questions of the IR task. Critical information needed to answer the first question is found at 385 s. (media time) while information needed to answer the last question is found around 3120 s. Thus, for this meeting we calculate an information span of IS = 3120 - 385 = 2735 s.

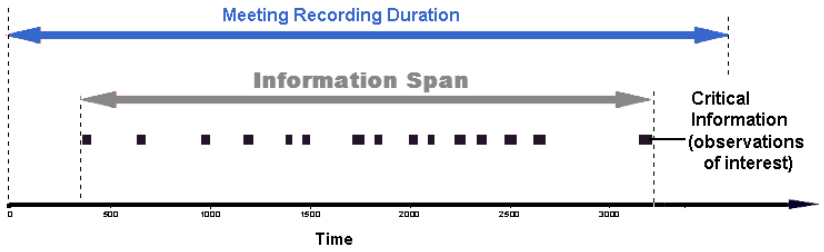


Figure 9.10: Critical Information Distribution and Information Span of evaluation meeting A

Note that a meeting's information span is entirely dependant on the information seeking task. As an example, consider a meeting in which a number of participants first spend an hour and a half discussing the pros and cons of a number of various options and the last half an hour making up an action plan. If the purpose of a subsequent information seeking task is to find out what were the decisions taken by the attendees, then the information span consists only of the meeting last half an hour (Figure 9.11b). However, if one were not so much interested in the decisions taken (i.e. information already known) but more in the argumentation which led to these decisions, as would be the case for an absentee for example, than the information span in this case shifts to the first hour and a half of the meeting (Figure 9.11a).

We define the Information Accessed Ratio (IAR) as the ratio of the span of the critical

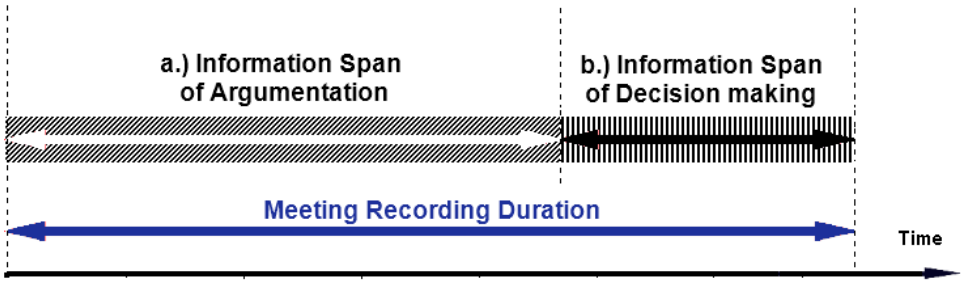


Figure 9.11: A meeting Information Span depends on the focus of the information retrieval task

information accessed by the user and efficiently used for purpose of the specific IR task (i.e. which translated in correct answers) over the meeting information span. Thus the information accessed ratio is equal to IAR $= \frac{User's\ Information\ Accessed\ Span}{IS}$. If the subject manages to answer all the questions correctly, his IAR is equal to 1. If he answers none, his IAR is equal to 0. In the exceptional case where the user only manages to answer a single question correctly, IAR can either be calculated as the ratio of relevant information needed to answer that specific question over meeting duration, or using a default listening granularity (e.g. 1 minute).

The IAR concept is illustrated in Figure 9.12: a user answered a number of questions during the task (including the second and up to the second-last), for which he had to access the critical information circled. As the information needed to answer the second question is found at 655 s. (media time) while information needed to answer the second-last question is found around 2597 s., this gives us an IAR equal to: $\frac{2597-655}{2735} = 71\%$.

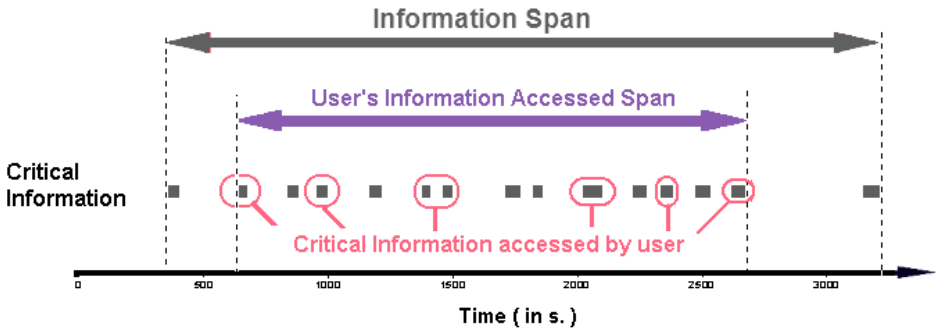


Figure 9.12: A User's Information Accessed Ratio of evaluation meeting A

The IAR metric reflects how *spread* is the information that was accessed by a user and thus can potentially reveal information about his browsing behaviour. As an example, consider Figure 9.13: in this case, the user has listened sequentially to 25 minutes of the meetings and although he managed to answer as many questions as the user of Figure 9.12, the distribution of the information which has been accessed is very different indeed: in the second example, the experiment participant has a very good knowledge of the first half of the meeting but

has no idea about what happens in the second half (IAR = $\frac{1470-385}{2735}$ = 39.7%). On the other hand, the participant in the first example may not have such a thorough knowledge of the first half of the meeting, but one would expect him to have a *wider* understanding of the meeting as a whole.

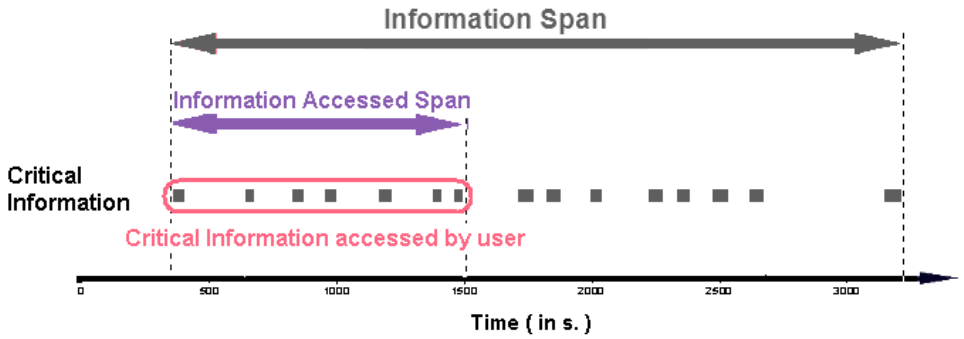


Figure 9.13: Sequential listening may result in relatively good score at the task but with low Information Accessed Ratio

Spanned Recall (SR): The motivation behind the IAR metric is to give an indication of the spread of information access. However, it is insufficient to measure a user's performance on its own. Consider the case where a user only answers the *first* and the *last* questions (media time). This would correspond to a IAR of 1, as the spread of the information accessed is equal to the meeting's IS, however, this is pretty bad performance as the recall would be $\frac{2}{15} = 13.3\%$, considerably worse than the participant illustrated in Figure 9.13, who although sequentially listened to the first 25 minutes of the meeting, managed to answer 7 out of 15 questions, thus a recall score of $\rho = \frac{7}{15} = 46.7\%$ In order to take both notions into consideration, we propose a novel metric called Spanned Recall and defined as the product of recall by the information accessed ratio.

Definition 9.1 Spanned Recall (SR)

The Spanned Recall is the product of a users's recall (ρ) by his Information Accessed Span (IAR):

$$\mathbf{SR} = \rho \times \mathbf{IAR} \qquad (\mathbf{SR} \in [\mathbf{0}, \mathbf{1}])$$

As an example, consider 2 participants who have answered the same number of questions (e.g. 50 % of task completed). The only difference is that the first participant sequentially accessed critical information in only half the meeting (IAR = .5), while the second one scored an IAR equal to 1, as he accessed critical information in the whole meeting (i.e. he answered the first and last question, media time). In this case, the first participant has a spanned recall of SR=.5 × .5= .25 while the second one has a SR=.5 × 1= .5. In other words, even though these two participants have

answered the same number of questions, the second one is rewarded for the spread of information access. Figure 9.14 illustrates the spanned recall with two examples of user browsing behaviour.

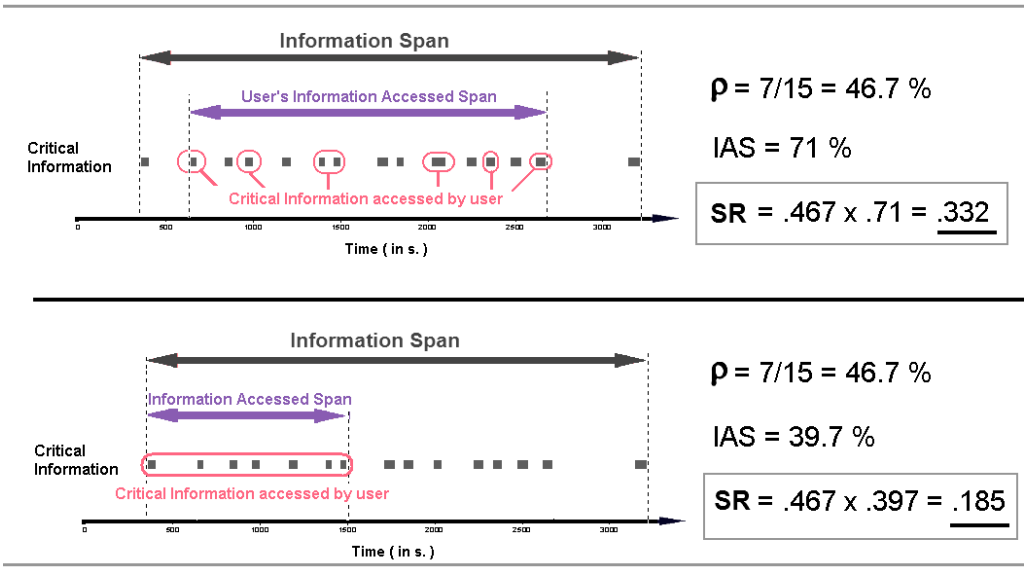


Figure 9.14: Spanned Recall for 2 types of browsing behaviour

9.4.4 Proposed Interpretation of Novel Metrics

Interpretation of First Correct Answer Metric

One could argue that the FCA can only be used as a performance measure if participants are aware that they need to provide a correct answer in the shortest possible time. Due to the nature of the IR task envisaged by the BET (to answer as many question as possible in a set amount of time), it will generally be the case for most users. However, certain participant could also decide to try to get an overall feel for the meeting before answering any specific questions and one can not rule out (as we have witnessed during our own IR experiment described later in this paper) that participants will use a wide variety of search strategies during an IR task.

For this reason, the FCA metric is not meant to be interpreted solely on its' own as a performance metric but in combination with an overall score metric such as the spanned recall (SR) The combined result of FCA and overall score, can provide some insight into a user's browsing strategy. A high FCA (long time to answer first question) coupled with a high overall score could be interpreted as a user waiting to gather sufficient information before answering the questions. In this scenario, a user successfully employs a strategy of information accumulation (listen first, answer later). At the other end of the spectrum is the user with a very low FCA (has very quickly identified some information of interest) and low overall score. In this case, it is likely that the user stumbled across the first piece of useful information by chance and that the system did not overall well support him in his information retrieval task.

Interpretation of Information Accessed Ratio

A meeting information span is not a metric *per se*, rather it is a normalising factor to calculate the information accessed ratio. So what does the information accessed ratio tell us? The first thing is how much of a meeting's information span was *not successfully used* for the purpose of the information seeking task (corresponding to $1 - IAR$). If a user has an IAR of 60%, then we can infer that the user was unable to successfully use critical information in the remaining 40% of a meeting information span. It does not however means that he managed to answer *all* the questions in the IAR of 60% but only some of them (the metric defined in the following section includes a measure of performance in the IR task). There could be a number of reasons why a user has not successfully managed to use critical information in certain sections of a meeting. The first is simply because the information was not accessed, maybe for lack of time, or because a user may have skipped over it (using random access). In this case, one can assume that either the browsing system did not support the user sufficiently enough to enable him to complete the task in time, or that it did not provide sufficient information feed-back to point the user to a potential useful information. Another possible reason is simply that the information was indeed accessed but not understood by the user or that he might not have realised it was relevant to the information seeking task. A possible way of distinguishing between these two cases (information not accessed vs. information not understood) would be to simply log the media time accessed by the user of the meeting browsing system during the IR experiment.

The other possible use of the IAR metric is to potentially reveal information about a user's browsing behaviour. As an example, consider Figure 9.13: in this case, the user has listened sequentially to 25 minutes of the meetings and although he managed to answer as many questions as the user of Figure 9.12, the distribution of the information which has been accessed is very different indeed: in the second example, the experiment participant has a very good knowledge of the first half of the meeting but has no idea about what happens in the second half (IAR $= \frac{1470 - 385}{2735} = 39.7\%$). On the other hand, the participant in the first example may not have such a thorough knowledge of the first half of the meeting, but one would expect him to have a *wider* understanding of the meeting as a whole.

Interpretation of Spanned Recall

The spanned recall metric does the difficult job of providing both a measure of success at the task (through recall) and span of information access (through IAR). Some could argue that by definition, the metric is biased towards random access (and thus a browser) at the expense of sequential access. However, as meeting recordings are essentially composed of time-based media, only relying on recall as a performance measure, even if combined with speed of answering as suggested in the BET, does not permit to distinguish between a wide variety of user performances and browsing behaviour. While we accept that the spanned recall metric has some shortcomings,

we believe it offers an improvement on previous approaches. Thus, we choose to use SR as an overall user performance measure of the success at the IR task.

9.4.5 Results of the IR Task

Recall & Precision

Recall and precision results under the four conditions (2 meetings x 2 interfaces) can be found in Table 9.4. Table 9.3 describes how the results are displayed: the participants 1 to 10 used the Meeting Miner interface with meeting A, and those results are displayed at the bottom left-hand quarter of the table (condition Meeting A + Meeting Miner). The same subjects used the RAI interface with meeting B, and their results are displayed at the top right-hand corner of the table.

Table 9.3: Display of results under the four experiment conditions

	Meeting A		**Meeting B**	
	Recall	Precision	Recall	Precision
RAI	**Participants 11-20**		**Participants 1- 10**	
Meeting Miner	**Participants 1- 10**		**Participants 11-20**	

The results of Table 9.4 show that overall, precision is good in all the experiment conditions and that they are even better with the Meeting Miner. This tells us two things: the first is that overall, participants did not try to guess the answers to the questions and only submitted an answer when they were reasonably confident they had heard the relevant information. As the remaining time for the experiment was not displayed on the QUI interface, we avoided the phenomenon of the last minutes surge in answers which are generally due to participants trying to guess as many answers as they can in the last moments of the experiment. The reasons why users of the Meeting Miner achieved better precision results seems to be due to the fact that keyword search and indexing functionalities made them aware of potential multiple answers (e.g. participants made a decision but later changed their mind) whereas in the RAI condition, participants generally considered the first piece of relevant information to be the correct answer (and ignored subsequent relevant information). This would be confirmed by anecdotal evidence from the experiment participants and described in the second usability study of the Meeting Miner, detailed later in this chapter (section 9.5).

Recall is better with the Meeting Miner than the RAI interface, and this is particularly true in the case with evaluation Meeting B (travel itinerary in China). With nearly 20% recall difference between the RAI and Meeting Miner conditions, anecdotal evidence from the experiment subjects suggest that participants found this meeting much more difficult: it was longer, with a lot of information about unfamiliar places, etc. Thus, it is an encouraging result to find that the Meeting Miner search and indexing functionalities proved even more useful in a difficult IR task. The difference in recall with the meeting A condition, even though still significant at 13.3%, would

Table 9.4: Recall And precision Results of IR task

	Meeting A		**Meeting B**	
	Recall	Precision	Recall	Precision
RAI	53.3	80	33.3	83.3
	40	85.7	6.7	50
	40	100	40	85.7
	53.3	100	33.3	100
	40	100	26.7	100
	6.7	25	33.3	75
	46.7	87.5	46.7	100
	40	85.7	20	75
	40	75	13.3	100
	40	75	46.7	100
Average	**40%**	**81.4%**	**30%**	**86.9 %**
Meeting Miner	53.3	100	66.7	100
	40	85.7	46.7	100
	66.7	100	66.7	100
	33.3	83.3	33.3	100
	47	100	53.3	88.8
	60	90	33.3	71.4
	66.7	100	46.7	87.5
	53.3	100	40	100
	40	85.7	53.3	100
	66.77	100	53.3	100
Average	**52.7%**	**94.5%**	**49.3 %**	**94.8 %**

suggest that if the content of a meeting recording is familiar to the users, they have less problem finding relevant information through conventional Random Access.

The bar chart in Figure 9.15 display the recall results according to individual participants, with Meeting A represented by the left bar and Meeting B represented by the right bar, and the distinction between the two interfaces conditions represented by two different colour codes. The first interesting fact which is highlighted by this chart is that in almost all cases, participants have achieved better recall with the Meeting Miner than with the RAI. The only exception to this is participant 14 who did better with the RAI than the Meeting Miner (53.3% vs. 33.3%), while participants 4, 17 and 18 achieved the same scores with both interfaces. The other interesting fact is that the difference between scores is less obvious for the participants 11 to 20. The reason for this, as previously mentioned, is that meeting A (organising concert in Dublin) was generally found to be easier, and contained information more familiar to users. As a result, participants found they could navigate the recording relatively well with standard random access functionalities. The converse was not true: with meeting B being much harder, the performance of subjects using the Meeting Miner was much higher in comparison to those who used the RAI to navigate this recording (participants 1 to 10).

We also observed a large difference in participants' ability to browse through the meeting

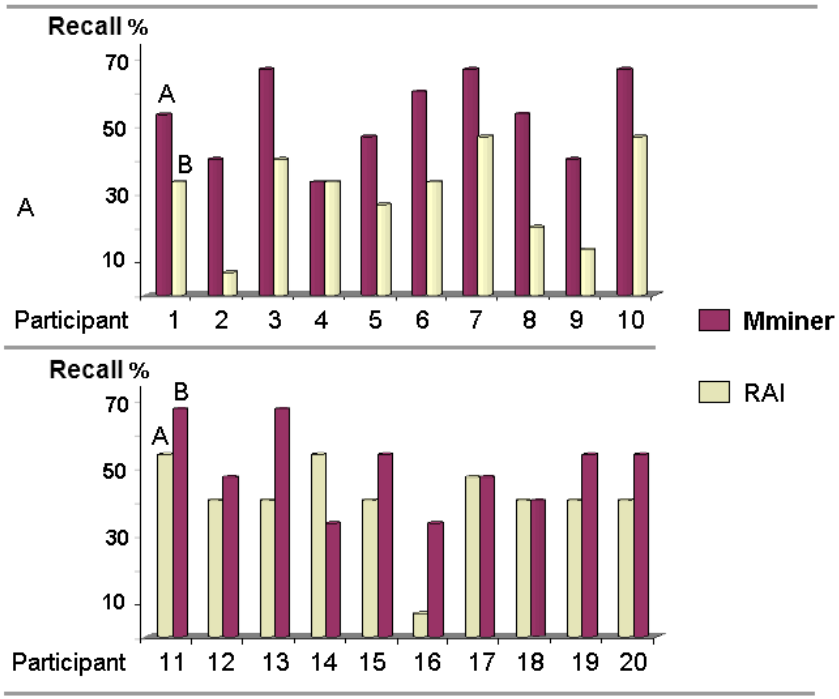


Figure 9.15: Recall according to Participants & interfaces' conditions, with Meeting A on the left and Meeting B on the Right

recordings and there is no doubt that some people are much better at identifying relevant information than others. Using the RAI on meeting A, participant 11 outperformed many of the subjects who used the Meeting Miner on the same recording. Interestingly, for participants who seemed to have had serious difficulties in locating relevant information and thus, with the lowest overall scores (participants 2, 9 and 16), the use of the Meeting Miner seems to have improved their performances.

Average Time & First Correct Answer

Table 9.5 shows the first correct answer-time and average answering time under the four conditions of the experiment. The figures would strongly suggest that the Meeting Miner contributed to a reduction of the time required to identify the first piece of relevant information: a reduction of $\frac{410-235}{410} = 42.7\%$ for meeting A and a dramatic reduction of $\frac{590-278}{590} = 52.9\%$ for meeting B. There are of course a couple of cases when participants seem to have stumbled across relevant information by chance: participants 14 who was the quickest to answer a question regarding meeting B with the Meeting Miner (98 s. or 1m. 38 s.) also was the slowest to answer subsequent questions (300 s. or 5m.) and went on to score one of the worst results in that experiment condition.

The phenomenon of information accumulation seems to have smoothed the advantage first provided by the search and indexing functionalities of the Meeting Miner in the case of meeting A, as we find an Avg. A.t only 16.5% lower than the RAI. However, in case of meeting B, participants

Table 9.5: First Correct Answer & Average Answering Time of IR task

	Meeting A		**Meeting B**	
	FCA-t (in s.)	Avg. A.t (in s.)	FCA (in s.)	Avg. A.t (in s.)
RAI	222	150	243	250
	450	215	463	750
	703	250	669	214
	564	188	563	300
	389	250	640	375
	213	375	881	375
	230	188	505	214
	443	223	617	375
	405	223	725	750
	489	188	596	214
Average	**410 s.** (6 m. 50 s.)	**225 s.** (3 m. 45 s.)	**590 s.** (9 m. 50 s.)	**382 s.** (6 m. 22 s.)
Meeting Miner	223	188	268	150
	133	215	562	214
	58	150	377	150
	269	250	98	300
	396	215	209	167
	370	150	356	214
	139	150	105	188
	201	188	347	250
	417	215	163	188
	146	150	296	188
Average	**235 s.** (3 m. 55 s.)	**187 s.** (3 m. 7 s.)	**278 s.** (4 m. 38 s.)	**201 s.** (3 m. 21 s.)

using the Meeting Miner achieved an Avg. A.t 47.4% ($\frac{590-278}{590}$) lower than the RAI. In terms of the number of questions answered, this translate in an average of 7 answers per participants using the RAI against 8.3 answers per participants using the Meeting Miner for meeting A, a difference of 1.3 questions over the 25 minutes task duration. On the other hand, participants using the RAI with meeting B answered on average 4.8 questions against 7.8 for those using the Meeting Miner: a difference of 3 questions on average or 62.5% more. This would tend to confirm that, in the case of a difficult meeting, it would be much harder for users to identify relevant information with random access alone and that the search and indexing functionalities provided by the Meeting Miner were indeed warranted.

Information Span & Spanned Recall

Table 9.6 displays the participants' information accessed ratio and spanned recall results. As participants 2 and 16 only managed to answer correctly one single question using the RAI (out of 4 questions answered), their results were exceptionally low and thus, not included in the average calculation of IAR and SR (outliers).

For meeting A, which has a IS of around 45 minutes and 35 seconds, participants using the RAI achieved an IAR of .633, which means that they accessed around 63.3% of the meeting information

Table 9.6: Information Accessed Ratio & Spanned Recall

	Meeting A		Meeting B	
	I. A. S.	S. P.	I. A. S.	S. P.
RAI	0.757	0.403	0.369	0.123
	0.537	0.215	-	-
	0.727	0.291	0.577	0.231
	0.879	0.469	0.368	0.123
	0.613	0.245	0.091	0.024
	-	-	0.091	0.03
	0.609	0.284	0.369	0.172
	0.493	0.197	0.199	0.04
	0.537	0.215	0.21	0.028
	0.542	0.217	0.368	0.172
Average	**0.633**	**0.282**	**0.294**	**0.105**
Meeting Miner	0.757	0.403	0.536	0.358
	0.727	0.291	1	0.467
	1	0.667	0.81	0.54
	0.493	0.164	0.89	0.296
	0.757	0.354	0.89	0.474
	0.879	0.527	0.367	0.122
	0.879	0.586	1	0.467
	0.613	0.369	0.443	0.177
	0.879	0.352	0.447	0.238
	1	0.667	0.89	0.474
Average	**0.798**	**0.438**	**0.544**	**0.361**

span. Using the Meeting Miner, participants achieved an IAR of 79.8%. As an example of the difference in information access between users of the 2 interfaces: only one user managed to answer the very last question (media time) using the RAI while 5 people answered this question using the Meeting Miner. Using the Meeting Miner, two participants (3 & 10) managed to answer both the first and last questions (media time) thus achieving an IAR of 1. Although, participant 9 achieved a ρ below average (40%), he had a good IAR (87.9%) which means that although he might have answered fewer questions, these were well spread over the information span of the meeting. The spanned recall, which takes into consideration both the recall and the information accessed ratio stands at .294 using the RAI and .438 using the Meeting Miner.

For meeting B, the difference between the results obtained with the two interfaces is once again even more pronounced. IAR stands at .294, which means that participants only successfully covered on average 29.4% of the useful information of the meeting. With the Meeting Miner, the IAR jumps to 72.7%. As an interpretation of this result, while none of the RAI users answered any of the task's last three questions (media time), 8 of 10 users of the Meeting Miner answered at least one or more of these questions. Participants 12 and 17 answered both the first and last questions (media time): considering that the meeting information span was 59 minutes 45 seconds, this time is considerably more than the duration of the experiment. The IAR results translate in a SR of .105 on average for the RAI users against a SR of .361 for the Meeting Miner users, a

threefold increase. The spanned recall of participants 1 to 20 can be seen on Figure 9.16.

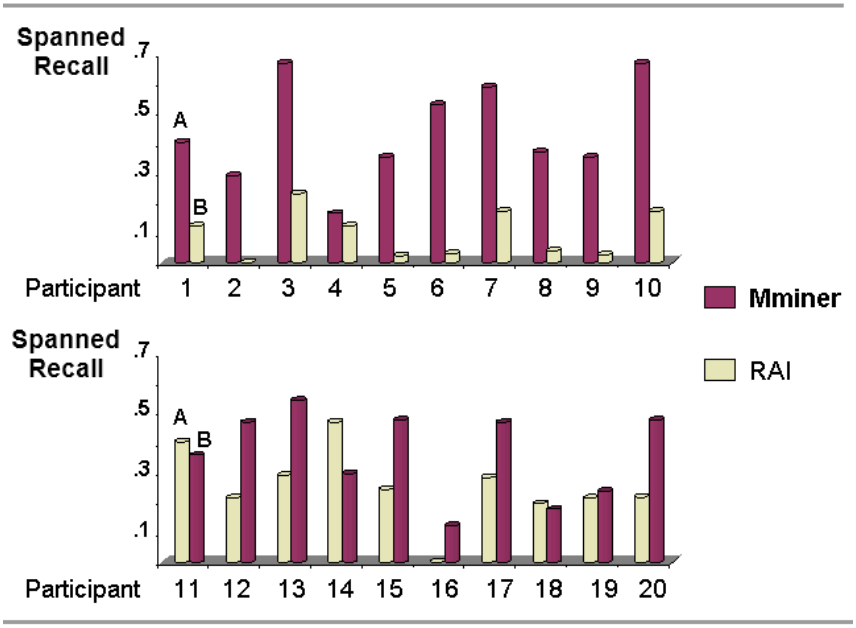


Figure 9.16: Spanned Recall according to participants & interfaces' conditions, with Meeting A on the left and Meeting B on the Right

9.4.6 Concluding Remarks on Information Retrieval Task

During, the IR task, we observed a wide diversity of performance across participants and to some extent, it was difficult at times to tell if the performance measured was that of the user or the system. There is no doubt that some people are much better at locating information quickly and efficiently, while some were struggling during the experiment. Some users using the RAI have outperformed users using the browsing tool. There were a number of encouraging results: the novel metrics of information accessed ratio and spanned recall seem to suggest patterns in which users of the Meeting Miner were able to have a broader access to meeting information. This was particularly pronounced in the case of the longer and harder meeting B. The other interesting point is that the people who had the most difficulties in locating information using the RAI interface displayed a much improved performance when using the Meeting Miner.

9.5 Final Usability Study

The next section describes the Meeting Miner final usability study, which was carried out in parallel with the IR evaluation task. There are two elements to this final usability study: *implicit* and *explicit* measurements of users preferences. The implicit element is described in the next section and consists in the log of users interactions with the interface. Explicit participant feed-

back was obtained through a usability questionnaire handed at the end of the IR task and informal discussion with the author.

9.5.1 Interactions with The Meeting Miner

Participants of the IR task described in the previous section used a version of the Meeting Miner which was modified in order to capture all their interactions with the interface: all keystrokes, mouse clicks, keywords and topic search selections were captured and timestamped. As a result, we obtained a clearer picture of participants' interactions with individual components of the system. This is illustrated by Figure 9.17. Once again, keyword (38.4%) and Topic search (19.1%) were the most used components of the Meeting Miner, and their combined usage accounted for more than half of all interactions with the Meeting Miner's interface. Once a keyword search was performed, most users simply listened sequentially to the speech segments selected by the system. However, a number of participants also used the Audio View panel, which displays speech according to participants and important index words, for local audio navigation (21.4%) in order to jump back and forth in the recording. Random audio access was used scarcely (7.6%) and so was the meeting document-based browsing (8.9%), as users generally found the documents too long to read (even though they were never longer than two A4 pages long) and preferred using the keyword list instead. Finally, the paragraph-view was the least popular component, with only 4.6% of interactions. The following section which details participants' explicit comments on the Meeting Miner system will shed some light on these patterns of system usage.

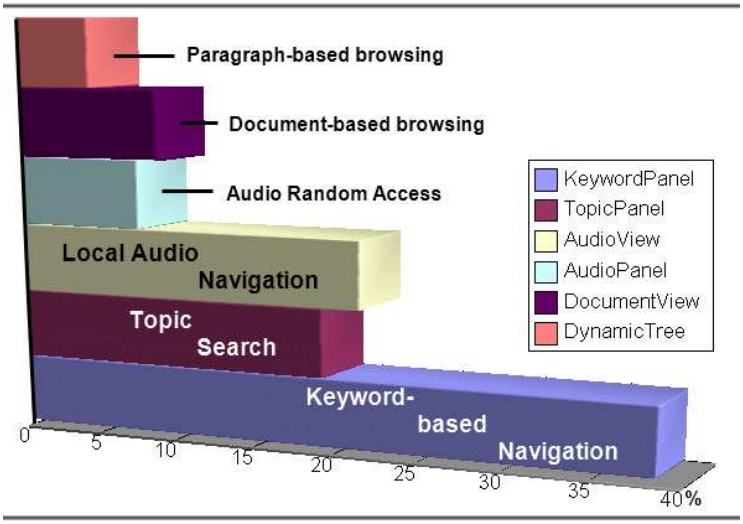


Figure 9.17: Participants' interactions with the various components of the Meeting Miner during the IR task

9.5.2 Participants' Comments

Participants comments were collected both as a written usability questionnaire and through informal discussion with the author. The following comments highlight some of the subjects' opinions about the evaluation experiment and the main differences between browsing the two meetings with the two interfaces.

About the Task

Participants generally found the IR task quite challenging. This confirms previous studies of audio browsing behaviour which concluded that users can have great difficulties in navigating audio recordings (Whittaker et al. 1998a,b, Nakatani et al. 1998). As previously mentioned on several occasions in this chapter, participants found meeting B much harder.

Question 9.5 How challenging was this task?

- *"Tasks were challenging, particularly due to length of meeting..."*
- *"Tasks were challenging because the answers were scattered all over the audio, with some topics discussed at different stages..."*
- *"There was a lot of information..."*
- *"many people going to many places in China so it required a lot of work"* vs. *"interesting to hear all the effort needed to put on a gig..."*
- *"First (meeting A) was easy, second (meeting B) much more difficult..."*

The next question explains why participants generally obtained very high precision scores in all experiment conditions: due to the fact that time was not displayed, there was no last-minute rush to answer questions and the multiple choice answers had been designed well enough to discourage guessing.

Question 9.6 What did you think of the Multiple Choice Answers?

- *"Some were very specific and required listening to several audio segments..."*
- *"Not easy enough to make a guess, interesting..."*
- *"difficult..."*
- *"well balanced to discourage guessing..."*
- *"as randomly ordered, made you concentrate on what you were listening to..."*
- *"plausible answers for all questions, so you couldn't guess easily.."*

About the Audio Recording

An interesting finding of the experiment was that most of the participants never really took the time to read the meetings document outcomes. Although these would have been less than two A4 pages, participants went straight to the keyword search and generally preferred listening to the audio recording. If this pattern was verified on a larger scale, it could have important implications in meeting browsing technology as regard the usefulness of displaying speech transcripts, even in a high recognition scenario: if users do not take the time to read two pages-long meeting minutes, what are the changes they will be reading the 24 pages of transcripts of a meeting (Appendix B)?

As confirmed by previous studies (Whittaker et al. 1998a,b, Nakatani et al. 1998), many participants found listening to the speech recordings quite challenging: they found it confusing and incoherent at time, with a lot of peripheral information only of marginal interest to the task. Several complained that meeting participants often changed abruptly the subject of conversation. Although many of the people who participated in the experiment found it interesting, mainly because they were curious about testing a new and unusual application, several participants found listening to someone else's conversations strange, and in some cases *"a bit tedious"*. Users' listening frustration was obviously compounded by the fact that the experiment was an artificial task in which they had no involvement. In a real-world scenario, an end-user meeting browsing tool application would primarily appeal to people with some direct interest in the meeting recordings' content, such as the meeting participants themselves or someone with definite reasons for reviewing the meetings' content.

Question 9.7 How useful was the audio recording?

- *"much easier than having to read the text constantly..."*
- *"very useful, as helped you put things in context..."*
- *"difficult... people manage to say nothing for minutes at a time and often have several partial conversations about a topic at different times, making it hard to find information"*
- *"audio is not a good medium for storing/browsing information!"*
- *"there was a lot of useless information..."*
- *"tedious and rambling..."*
- *"dull..."*

Several people found the task involving the RAI frustrating because of the lack of reference points which resulted in users having the impression of listening to *"random"* excerpts of speech. Users feed-back confirmed that the Meeting Miner search and indexing functionalities certainly helped navigation. However, some still found frustrating to have to listen to a certain amount of

audio recording before being able to identify the correct information. Some suggested adding more advanced skimming facilities on top of the existing search functionalities.

Question 9.8 How did you find navigating the audio recording?

- *"very easy... really liked the point & click feature, allowed me to play different part of audio at will..."*
- *"colour coding the participants helped..."*
- *"simple, selecting audio was easy, search was pretty useful..."*
- *"...could see when each person was speaking and could skip the silence parts of the audio... keywords were helpful..."*
- *"...keyword search was useful. Helped very much in guiding the search for the answer..."*
- *"skipping between segments can be confusing..."*
- *"I found it difficult to locate information..."*

RAI vs. Meeting Miner

Browsing the meeting recordings with the two different interfaces was indeed a very different experience for the users. While users felt that finding information with the RAI was *"random"* or down to *"luck"*, the keyword and topic search functionalities were confirmed to be the primary and most convenient way to quickly identify relevant sections of the meeting recordings.

A More Thorough Search

A very interesting remark concerning the difference in approach to navigating the meeting recording with the Meeting Miner or the random access interface was made by several participants: users were more confident about their answers when using the Meeting Miner (which was vindicated by the higher precision scores). When using the RAI, the first piece of information the users came across in relation to a specific query was generally assumed to provide the correct answer. In the Meeting Miner, as various information locations may have possibly been highlighted in the Audio View, users were more aware of potential alternative information locations. As a result, several users reported being more thorough in answering a specific question, listening to several speech segments before eventually deciding that they had definitely heard the correct information.

Question 9.9 What are the main differences between browsing the meetings with the Meeting Miner and the Random Access Interface?

- *"browsing (with RAI) is almost entirely random. Finding information with (Meeting Miner) browser is a little easier..."*
- *"looking for information with (RAI) is random, making it difficult to find specific information..."*
- *"it is easier to get to a particular topic with the (Meeting Miner) browser... difficult to find the answer to a particular question (with RAI)..."*
- *"(RAI) is difficult because finding the answers is a lot to do with luck. The (Meeting Miner) browser allows for more intelligent search..."*
- *"it is easier to find specific sections with the (Meeting Miner), with the (RAI) you can only really search forward through thin air..."*
- *"with the Meeting Miner browser, you're able to pick out certain words which may help you locate a specific comment you'd like to hear..."*
- *"search features. The ability to combine search terms was very useful, also listing their frequency"*
- *"the ability to search for keywords is the primary advantage (Meeting Miner)"*
- *"being able to search, much more efficient searching with the (Meeting Miner) browser... had to listen to nearly all dialog to find information needed (with RAI)..."*
- *"with RAI, I was less inclined to verify my answers..."*
- *"less random sampling (with Meeting Miner), difficult to find information you are looking for (with RAI)..."*
- *"Meeting Miner gives a clue about the content of the audio or topics under discussion..."*
- *"No idea where to click, no visual clues (with RAI)..."*
- *"more efficient browsing (with Meeting Miner), easier to brake down in specific topics, less tedious than (RAI) "*
- *"(the RAI) it's crap..."*

9.5.3 Strength & Limitations of Current System

Strength of Current System

Users made a lot of positive comments about the Meeting Miner functionalities and the results of the IR task showed that users of the Meeting Miner generally outperformed users of the RAI,

particularly in the experiment involving the harder and longer meeting B. Navigation feedback it terms of audio wear-marks and flexible indexing of keywords on the time line certainly improved navigation difficulties traditionally encountered by users of speech browsing systems. Keyword search permitted identifying potential areas of interest much quicker and the ability to combine words in Topic Search was sometimes very efficient in finding precisely the appropriate information.

Limitations: High Expectations of Browsing System

Users' expectations of the Meeting Miner system were sometimes greater than its' performance. For keyword and Topic search, some of the users relied entirely on segments selected by the system. They assumed the search would return segments with near perfect boundaries with relevant information highlighted and irrelevant information excluded. In practice, the information required to answer a question was often either in a middle of speech segments containing a significant amount of irrelevant information, and was occasionally nearby a given selection (but not highlighted). Sometimes, due to speech recognition errors or the time-based search limitations, the system either did not return any selections for specific queries or returned irrelevant segments. When there was no selections returned, some users wrongly assumed the information was not present in the recording. When the system returned irrelevant segments, users ended up being confused. Some understood it as a complete alternative to random access. In order to mitigate the over-reliance on the browsing functionalities of the system in the future, it should be made clear to the users that it only provides a *best-effort* solution and that local audio navigation might still be necessary in some cases.

Question 9.10 How would you rate the Meeting Miner system overall?

- *"the keyword and topic search were sometimes helpful but didn't always recover what you were looking for..."*
- *"(keyword and topic) search were not always perfect but helped in finding the relevant sections..."*
- *" certain words could be used several times, in irrelevant areas to what you wanted..."*
- *"speech navigation was a little confusing when a keyword search yielded multiple speech points..."*
- *"relevant audio clips are often outside those highlighted... finding information is very hit-an-miss..."*
- *"search bounds were not precise... needed to listen to audio before search recommendation..."*

Possible improvements

In the usability questionnaire, users were encouraged to suggest possible improvements to the Meeting Miner system. The following describes a number of theses suggestions.

Question 9.11 What do you think could improve the Meeting Miner?

- *"better representation of spoken content..."*
- *"finer granularity of (speech) sections, sometimes quite large and irrelevant..."*
- *"to search for terms not available (in keyword list)..."*
- *"ability to type own keywords..."*
- *"user defined keyword search..."*

As had already been highlighted by the previous heuristic and usability evaluations, one of the main limitation of the system was perceived as being it's limited search vocabulary. However, the primary motivation behind the use of a controlled vocabulary was mainly an effort to guarantee reasonably good results. Recall from the pilot usability study described in section 9.3 that as the quality of ASR transcripts were very poor, users just ignored them. The difficulty here would be to provide flexibility of queries without compromising the quality of the retrieval results. A possible solution to this problem and improvement to the current system is illustrated in Figure 9.18. The current retrieval mechanism is illustrated in Figure 9.18a.: the user can only formulate a limited number of queries, using the Controlled Vocabulary $\mathcal{V}$. A more flexible alternative is illustrated by Figure 9.18b.: in this system, the user can formulate any type of queries. Using a Thesaurus, similarities between the user query and the controlled vocabulary $\mathcal{V}$ are analysed and a number of potential reformulated queries (including the terms from $\mathcal{V}$) are proposed to the user. If one of these queries is deemed an acceptable alternative to the primary query (YES case), the retrieval system pursue the search. Otherwise, the user reformulates a different query.

9.6 Concluding Remarks on the Meeting Miner Evaluation

This chapter concludes our evaluation of the Meeting Miner browsing tool. While the analytic evaluation performed in the previous chapter enabled us to measure the retrieval system in terms of standard IR metrics, we presented in this chapter a number of experiments in which users are put back at the center of the evaluation process. The heuristic evaluation, performed early in the system life-cycle highlighted a number of usability issues, mainly in terms of audio navigation feed-back and functionalities and status of the search performed. The Meeting Miner was subsequently modified to take into considerations the most important of these usability issues. A pilot usability study gave

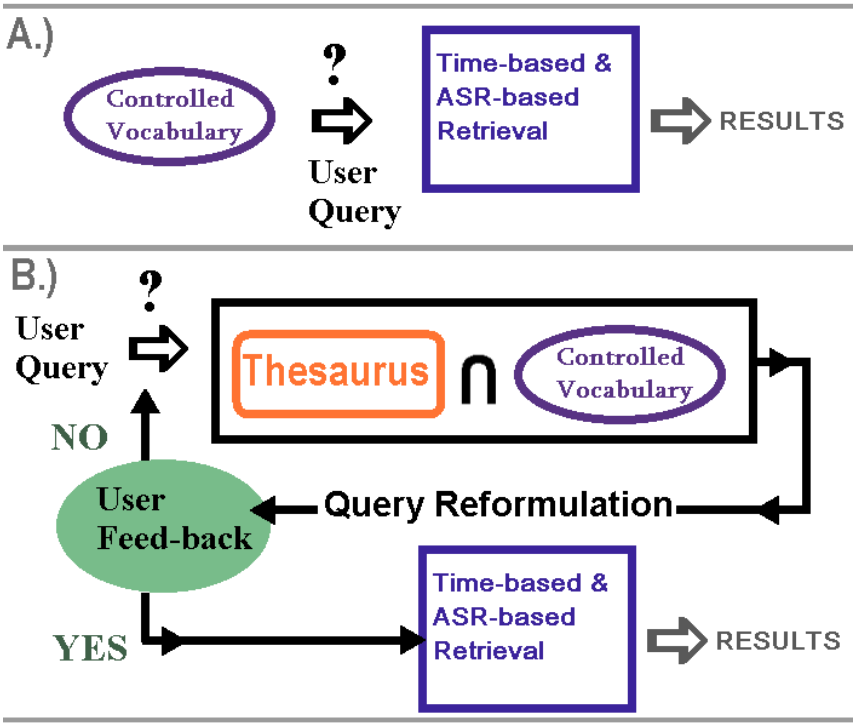


Figure 9.18: Query Reformulation Mechanism using a Thesaurus, Controlled Vocabulary & User feed-back

us additional user feed-back. It provided information about the components and functionalities of the Meeting Miner which were judged the most useful and some insight into users' browsing behaviours. Finally, it prepared the ground for the IR evaluation test, which compared the Meeting Miner against a random access interface in a variety of meeting conditions. Interestingly, while the performance of users of the Meeting Miner was only marginally better in the case of meeting A (easier and with familiar information), the additional indexing and search functionalities of the Meeting Miner proved critical in the experiment involving the second harder meeting B. It also seems to have helped users with the greatest browsing difficulties, although the small number of participants involved in the experiment prohibits us of making definite conclusions on this matter. Finally, the most serious limitations of the systems were identified as being a possible over-reliance of the result of the search functionalities. We do not believe this to be a major issue, as users would probably become familiar with the system limitations over time and develop more realistic expectations of its' performance. A more serious usability issue was identified as the limitations of the controlled vocabulary which was a persistent complaint made by users. We briefly described how one could introduce more flexibility in the querying process while trying to minimise the degradation of the system's performance. This functionality would however require a whole new set of evaluation experiments in order to measure it's impact on the retrieval process and as a result we will not discuss it any further.

Chapter 10

Conclusion

10.1 Validation of Hypothesis

The main motivation behind the research work presented in this thesis was the fact that developers of meeting browsers had generally attempted to adapt and combine techniques developed in multidisciplinary fields (MIR, NLP) while often overlooking a key aspect of meetings which is inherently present in participants' interactions. Our work focused specifically on using non-verbal interactions performed on shared meeting artefacts during online meetings to provide post-access to meeting recording information.

This thesis provided a validation of this approach through the development of the Meeting Miner, an interaction-based meeting browsing tool. The Meeting Miner was iteratively evaluated and improved through a combination of analytic evaluation and a multi-stage user evaluation.

Using non-verbal interaction metadata to access speech content proved to be a promising technique, albeit one which needs to be complemented by other technologies. ASR was used in the system to address issues of (time-based) sparsity of actions, while query expansion seem to be warranted for issues relating to the limited expressiveness which can be achieved through the sole use of interaction content. The main advantage of the approach however was that the content of non-verbal interaction exhibited a high level of semantic similarity with the content of concurrent speech and thus could be used with a high degree of reliability. The content of individual actions was efficiently used to navigate meeting recordings intuitively by most users. Aggregation of information through co-occurrence and co-location of actions (paragraph-based retrieval) was somehow far less successful. We identify two main reasons for this. The first is that this information representation was difficult for users to comprehend. They generally seemed more comfortable with more traditional information concepts such as the linear and sequential nature of speech, and query-based random access to speech content. The second issue was that aggregating information through co-occurrence and co-location of interaction does not always translate in semantic homo-

geneity due to the fast and often arbitrary nature of interaction in meetings. Further analysis, understanding and modelling of the nature of interaction in meetings are thus required in order to improve these techniques.

10.2 Lessons Learned

In Chapter 5, we showed how we propose to integrate traditional editing and awareness functionality in a remote meeting environment with mechanisms designed to implicitly capture document history. Managing in real-time the log of actions on semantic entities (paragraphs of text) was not a trivial problem, as even simple and standard operations on a text document can lead to a large variety of outcomes. We thus associated specific cases of text operations (e.g. Cut and Paste) with manipulation rules regarding paragraphs' interaction logs, in order to coherently manage this information in real-time. In certain cases, due to the real-time constraints of the remote architecture, compromises had to be found between the interaction logging requirements and the needs to preserve good responsiveness of the remote meeting system.

In Chapter 7, we described how we proposed to use the interaction metadata collected during the course of meetings to later provide post-meeting access to the recordings. We described how the content of actions could be used to implement query-term (keyword and topic) search as well as alternative navigation modality, such as action-based navigation and paragraph temporal neighbourhood retrieval. The main difficulties encountered here were how to use and organise the rich interaction information for information access without falling in the classic pitfalls of information overload. In order to limit the number of all potential actions or speech events to *meaningful* ones, several steps were taken:

buffering this step was performed during meeting capture, as the recording server buffered atomic operations (character-based) into more comprehensive ones (word or sentenced-based)

filtering this post-meeting processing step ensures that stop-words and non-vocabulary words were removed from the Meeting Miner's Vocabulary

combining overlapping speech segments into more comprehensive ones due to the large incidence of acknowledgements in remote meeting, with little semantic interest when taken out of context. Removing redundancy in information displayed (e.g. consecutive actions by same agent, etc).

The analytic evaluation performed in Chapter 8 proved beyond doubt a number of information assumptions and conceptual models used in the design of the Meeting Miner(Chapter 7). The high Precision of the Meeting Miner Keyword-based search vindicated the claims of *semantic overlap* between the content of text-event and concurrent speech turns, thus showing that text-events in meetings can indeed be used very reliably to access speech. This technique has of course some

serious limitations, both in terms of the extent of the potential vocabulary (limited to words which occur in text-events) and accessibility to recordings, limited to the set of text-events. We argued that these "weaknesses" are in fact the very advantages of these techniques:

(i) the limited vocabulary is likely to be a condensed version of the most-important meeting keywords (i.e. as opposed to all the words uttered during the meeting)

(ii) text-events are sparse and associated with important meeting events (e.g. a decision has been taken, critical information such as time-tables, plans, figures are recorded, a problem has been highlighted, etc.) and thus provide index to significant events of the meeting.

The analytic evaluation also highlighted important potential implications of psycho-linguistic phenomena for using text actions for meeting browsing. Repetitions in discourse and linguistic convergence showed that even a limited number of text-events could point to keyword and topic clusters in meeting recordings.

We presented a multi-stage evaluation of the Meeting Miner meeting browser, each step resulting in improvement of the functionalities of the system and the user interface. The user evaluation confirmed previous studies which highlighted the difficulties and frustration people have when navigating speech information. Important lessons were learned regarding what functionalities could enhance their browsing experience, the level of information feed-back required and the importance of flexibility in the browsing tool.

We used, and complemented with novel metrics, the Browser Evaluation Test proposed by Wellner et al. (2005). We found that users of the Meeting Miner outperformed users of a random access interface, and that this difference in performance was particularly important in the case of a longer and more complex meeting. This is a very encouraging result which highlighted the usefulness of the Meeting Miner as a browsing tool in accessing complex information.

10.3 Future Research

As this work represents one of the first comprehensive study of the use of low-level interaction information in order to provide access to meeting recordings, a vast amount of research remains to be done in order to unleash the full potential of interaction information as an information retrieval and navigation modality. We here briefly discuss some of the most important avenues for future research.

Meeting History Modeling: the history models presented in this thesis in Chapter4 and Chapter5 were intended to be generic models. We envisage these models to be perfected as our understanding of participants' interactions improves. In particular, we envisage that a large collection of meetings will highlight various trends in meeting interactions. Our intuition is different that type of interactions will impact on the information distribution in

meetings. Thus, better understanding how collaboration took place (e.g. tight collaboration v.s. loose, etc.) will provide more efficient ways of accessing information in meeting recordings.

Transpose the Model: There are two ways in which the generic interaction model presented in this thesis can be transposed to other fields of research. The first one would be to apply it to different types of space-based meeting artefacts. We envisage interaction-based techniques to be particularly well suited for highly *visual* information, like the ones typically used in collaborative design. As a result, it would be particularly interesting to implement interaction-logging and retrieval in a CAD (Computer Aided Design) environment, such as the one suggested in Chapter4 of this thesis (i.e. technical drawing, or architectural design). The second way in which this model can be transposed would be to implement it in a meeting room environment, as suggested by the work of Lalanne et al. (2004).

Higher-level Meeting Representations: Although we did attempt to provide alternatives to keyword-based queries in the Meeting Miner in the form of paragraph neighbourhood and action-based browsing, these techniques were met in practice with mixed success, partially because these indexing concepts seemed unfamiliar to users. We envisage future meeting browsing system to provide higher-level meeting representations, such as the ones currently investigated by McCowan et al. (2005), who have modelled meetings as a continuous sequence of high-level meeting group actions (monologue, presentation, discussion, etc.)

Meeting Browsers Evaluation Methods: we have highlighted the current lack of widely adopted methodology for evaluating meeting browsers and have discussed the commendable efforts made in that direction by Wellner et al. (2005)'s proposal of a generic Browser Evaluation Test and AMI's public release of a meeting corpus (Carletta et al. 2005). It was our ambition to contribute to the development of generic meeting browsers evaluation methods by proposing to complement the BET with novel evaluation metrics. As the area of meeting browser research matures, we expect evaluation methodologies to play a prominent role. One interesting avenue for improvement would be to complement the analytic aspect of a Browser Evaluation Test with widely adopted performance metrics and a usability methodology including a meeting browser set of usability heuristics.

Bibliography

Aigrain, P., Zhang, H. and Petkovic, D.: 1996, Content-based representation and retrieval of visual media: A state-of-the-art review, *Special Issue on Representation and Retrieval of Visual Media in Multimedia Systems*, Vol. 3, Kluwer Academic Publishers, pp. 179 – 202.

Allan, J., Carbonell, J., Doddington, G., Yamron, J. and Yang, Y.: 1998, Topic detection and tracking pilot study: Final report, *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*, Virginia, US.

Allen, J. F.: 1983, Maintaining knowledge about temporal intervals, *Communications of the ACM* **11**(26), 832–843.

AMI Corpus: 2006, multi-modal data set, http://www.idiap.ch/amicorpus.

AMI Project: 2006, *Augmented Multi-party Interaction*, http://www.amiproject.org/.

Anupam, V. and Bajaj, C. L.: 1994, Shastra: Multimedia collaborative design environment, *IEEE MultiMedia* **1**(2), 39–49.

Arons, B.: 1992, Techniques, perception, and applications of time-compressed speech, *Proceedings of Conference of American Voice I/O Society (AVIOS)*, pp. 169–177.

Arons, B.: 1997, SpeechSkimmer: a system for interactively skimming recorded speech, *ACM Transactions on Computer-Human Interaction*, Vol. 4, ACM Press, New York, NY, US, pp. 3–38.

Bafoutsou, G. and Mentzas, G.: 2002, Review and functional classification of collaborative systems, *International Journal of Information Management*, Vol. 22, pp. 281–305.

Bertino, E. and Ferrari, E.: 1998, Temporal synchronization models for multimedia data, *IEEE Transactions on Knowledge and Data Engineering* **10**(4), 612–631.

Blakowski, G. and Steinmetz, R.: 1996, A media synchronization survey: reference model, specification, and case studies, *IEEE Journal on Selected Areas in Communications* **14**(1), 5–35.

Boreczky, J., Girgensohn, A., Golovchinsky, G. and Uchihashi, S.: 2000, An interactive comic book presentation for exploring video, *Proceedings of the SIGCHI Conference on Human factors in Computing Systems, CHI '00, The Hague, Netherlands*, ACM Press, New York, NY, US, pp. 185–192.

Bos, N., Olson, J., Gergle, D., Olson, G. and Wright, Z.: 2002, Effects of four computer-mediated communications channels on trust development, *Proceedings of the SIGCHI Conference on Human factors in Computing Systems, CHI '02, Minneapolis, Minnesota, US*, ACM Press, New York, NY, US, pp. 135–140.

Bouamrane, M.-M., King, D., Luz, S. and Masoodian, M.: 2004, A framework for collaborative writing with recording and post-meeting retrieval capabilities, *Special issue on the 6th International Workshop on Collaborative Editing Systems, CSCW 04, Chicago, US, IEEE Distributed Systems Online* .

Bouamrane, M.-M. and Luz, S.: 2006a, Navigating multimodal meeting recordings with the Meeting Miner, *in* H. Larsen, G. Pasi, D. Ortiz-Arroyo, T. Andreasen and C. Henning (eds), *Proceedings of Flexible Query Answering Systems, FQAS'2006, Milan, Italy*, Vol. LNCS 4027 / 2006, Springer Berlin/Heidelberg, Germany, pp. 356–367.

Bouamrane, M.-M. and Luz, S.: 2006b, Temporal mining of recorded collaborative production of artefacts, *Proceedings of Industrial Conference on Data Mining, ICDM'06*, Leipzig, Germany, pp. 187–201.

Bouamrane, M.-M. and Luz, S.: 2007a, An analytical evaluation of search by content and interaction patterns on multimodal meeting records, *Multimedia Systems Journal, Semantic Media Adaptation and Personalization Special Issue* **13**(2), 89–102.

Bouamrane, M.-M. and Luz, S.: 2007b, Meeting browsing, a state-of-the-art review, *special issue of Multimedia Systems Journal, User-Centered Multimedia* **12**(4-5), 439–457.

Bouamrane, M.-M., Luz, S., Masoodian, M. and King, D.: 2005, Supporting remote collaboration through structured activity logging, *in* G. C. F. Hai Zhuge (ed.), *Proceedings of 4th International Conference in Grid and Cooperative Computing, GCC 2005, Beijing, China*, Vol. LNCS 3795 / 2005, Springer Berlin / Heidelberg, Germany, pp. 1096–1107.

Brotherton, J. A., Bhalodia, J. R. and Abowd, G. D.: 1998, Automated capture, integration, and visualization of multiple media streams, *Proceedings of the IEEE International Conference on Multimedia Computing and Systems, ICMCS '98, Austin, Texas, US*, IEEE Computer Society, p. 54.

Buchanan, C. and Zellweger, P. T.: 1992, Specifying temporal behavior in hypermedia documents, *Proceedings of the ACM Conference on Hypertext, ECHT '92, Milano, Italy*, ACM Press, New York, NY, US, pp. 262–271.

Carbonell, J. and Goldstein, J.: 1998, The use of MMR, diversity-based reranking for reordering documents and producing summaries, *Proceedings of the 21st International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '98, Melbourne, Australia*, ACM Press, New York, NY, US, pp. 335–336.

Carletta, J., Ashby, S., Bourban, S., Flynn, M., Guillemot, M., Hain, T., Kadlec, J., Karaiskos, V., Kraaij, W., Kronenthal, M., Lathoud, G., Lincoln, M., Lisowska, A., McCowan, I., Post, W., Reidsma, D. and Wellner, P.: 2005, The AMI meeting corpus: a pre-announcement, *Proceedings of 2nd Workshop on Multimodal Interaction and Related Machine Learning Algorithms*, Royal College of Physicians, Edinburgh, Scotland.

Chen, F. and Withgott, M.: 1992, The use of emphasis to automatically summarize a spoken discourse, *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP-92, San Francisco, CA, US*, Vol. 1, pp. 229–232.

Chiu, P., Boreczky, J., Girgensohn, A. and Kimber, D.: 2001, LiteMinutes: an Internet-based system for multimedia meeting minutes, *Proceedings of the 10th International Conference on the World Wide Web, WWW '01, Hong Kong*, ACM Press, New York, NY, US, pp. 140–149.

Chiu, P., Kapuskar, A., Reitmeier, S. and Wilcox, L.: 1999, NoteLook: taking notes in meetings with digital video and ink, *Proceedings of the Seventh ACM International Conference on Multimedia (Part 1), MULTIMEDIA '99, Orlando, FL, US*, ACM Press, New York, NY, US, pp. 149–158.

Chiu, P., Kapuskar, A., Wilcox, L. and Reitmeier, S.: 1999, Meeting capture in a media enriched conference room, *Proceedings of the Second International Workshop on Cooperative Buildings, Integrating Information, Organization and Architecture, CoBuild '99, Pittsburgh, US*, Vol. LNCS 1670 / 1999, Springer Berlin Heidelberg, Germany, pp. 79–88.

Choi, J., Hindle, D., Pereira, F., Singhal, A. and Whittaker, S.: 1999, Spoken content-based audio navigation (SCAN), *in Proceedings of the International Congresses of Phonetic Sciences, ICPhS-99*, San Francisco , USA.

Cimino, J. J.: 1998, Desiderata for controlled medical vocabularies in the twenty-first century, *Methods of Information in Medicine* **37**, 394 – 403.

COWRAT: 2006, http://cowrat.berlios.de/.

Cutler, R., Rui, Y., Gupta, A., Cadiz, J., Tashev, I., wei He, L., Colburn, A., Zhang, Z., Liu, Z. and Silverberg, S.: 2002, Distributed meetings: a meeting capture and broadcasting system, *Proceedings of the tenth ACM International Conference on Multimedia, MULTIMEDIA '02, Juan-les-Pins, France*, ACM Press, New York, NY, US, pp. 503–512.

Dharanipragada, S. and Roukos, S.: 2002, A multistage algorithm for spotting new words in speech, *IEEE Transactions on Speech and Audio Processing* **10**(8), 542–550.

Dourish, P. and Bellotti, V.: 1992, Awareness and coordination in shared workspaces, *Proceedings of the 1992 ACM Conference on Computer Supported Cooperative Work, CSCW '92, Toronto, Canada*, ACM Press, New York, NY, US, pp. 107–114.

Ellis, C. A. and Gibbs, S. J.: 1989, Concurrency control in groupware systems, *Proceedings of the 1989 ACM SIGMOD International Conference on Management of Data, SIGMOD '89, Portland, Oregon*, ACM Press, New York, NY, US, pp. 399–407.

Ellis, C. A., Gibbs, S. J. and Rein, G.: 1991, Groupware: some issues and experiences, *Communications of the ACM* **34**(1), 39–58.

Erol, B., Lee, D.-S. and Hull, J. J.: 2003, Multimodal summarization of meeting recordings, *Proceedings of International Conference on Multimedia and Expo, ICME '03, Menlo Park, CA, US*, Vol. 3, pp. 25–28.

Erol, B. and Li, Y.: 2005, An overview of technologies for e-meeting and e-lecture, *IEEE International Conference on Multimedia and Expo, ICME'05*, IEEE press, Amsterdam, Netherlands, pp. 1000–1005.

Foote, J.: 1999, An overview of audio information retrieval, *ACM Multimedia Systems*, Vol. 7, pp. 2–10.

Furui, S.: 1999, Automatic speech recognition and its application to information extraction, *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics, Morristown, NJ, US*, Association for Computational Linguistics, pp. 11–20.

Furui, S.: 2003, Robust methods in automatic speech recognition and understanding, *Proceedings of EUROSPEECH, Geneva, Switzerland*, Vol. III, pp. 1993–1998.

Gall, D. L.: 1991, MPEG: a video compression standard for multimedia applications, *Communications of the ACM* **34**(4), 46–58.

Garofolo, J. S., Voorhees, E. M., Auzanne, C. G. and Stanford, V. M.: 1999, Spoken document retrieval: 1998 evaluation and investigation of new metrics, *Proceedings of ESCA ETRW on Accessing Information in Spoken Audio, Cambridge, U.K.*, pp. 1–7.

Geyer, W., Richter, H. and Abowd, G. D.: 2003, Making multimedia meeting records more meaningful, *Proceedings of International Conference on Multimedia and Expo, ICME '03, Menlo Park, CA, US*, Vol. 2, pp. 669–672.

Geyer, W., Richter, H., Fuchs, L., Frauenhofer, T., Daijavad, S. and Poltrock, S.: 2001, A team collaboration space supporting capture and access of virtual meetings, *Proceedings of the 2001 International ACM SIGGROUP Conference on Supporting Group Work, GROUP '01, Boulder, Colorado, US*, ACM Press, New York, NY, US, pp. 188–196.

Gibbs, S., Breiteneder, C. and Tsichritzis, D.: 1994, Data modeling of time-based media, *Proceedings of the 1994 ACM SIGMOD International Conference on Management of Data, SIGMOD '94, Minneapolis, US*, ACM Press, New York, NY, US, pp. 91–102.

Goldman, J., Renals, S., Bird, S., de Jong, F., Federico, M., Fleischhauer, C., Kornbluh, M., Lamel, L., Oard, D., Stewart, C. and Wright, R.: 2005, Accessing the spoken word, *International Journal of Digital Libraries* **5**(4), 287 – 298.

Greenberg, S. and Marwood, D.: 1994, Real time groupware as a distributed system: concurrency control and its effect on the interface, *CSCW '94: Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work, Chapel Hill, North Carolina, US*, ACM Press, New York, NY, US, pp. 207–217.

Gutwin, C. and Greenberg, S.: 1999, The effects of workspace awareness support on the usability of real-time distributed groupware, *ACM Trans. Comput.-Hum. Interact.* **6**(3), 243–281.

Gutwin, C., Roseman, M. and Greenberg, S.: 1996, A usability study of awareness widgets in a shared workspace groupware system, *Proceedings of the 1996 ACM Conference on Computer Supported Cooperative Work, CSCW '96, Boston, Massachusetts, US*, ACM Press, New York, NY, US, pp. 258–267.

Hanjalic, A.: 2003, Generic approach to highlights extraction from a sport video, *Proceedings of International Conference on Image Processing, ICIP 2003, Barcelona, Spain*, Vol. 1, IEEE press, pp. 1–4.

Hearst, M. A.: 1994, Multi-paragraph segmentation of expository text, *Proceedings of the 32nd Annual Meeting on Association for Computational Linguistics*, Morristown, NJ, US, pp. 9–16.

Hersh, W., Pentecost, J. and Hickam, D.: 1996, A task-oriented approach to information retrieval evaluation, *Journal of American Society on Inormation Science* **47**(1), 50–56.

Hill, W. C., Hollan, J. D., Wroblewski, D. and McCandless, T.: 1992, Edit Wear and Read Wear, *Proceedings of the SIGCHI Conference on Human factors in Computing Systems, CHI '92, Monterey, California, US*, ACM Press, New York, NY, US, pp. 3–9.

Hindus, D. and Schmandt, C.: 1992, Ubiquitous audio: capturing spontaneous collaboration, *Proceedings of the 1992 ACM Conference on Computer Supported Cooperative Work, CSCW '92*, ACM Press, New York, NY, US, pp. 210–217.

Hirschberg, J., Whittaker, S., Hindle, D., Pereira, F. and Singhal, A.: 1999, Finding information in audio: a new paradigm for audio browsing and retrieval, *in* I. Mani and M. T. Maybury (eds), *Proceedings of the ESCA workshop: Accessing Information in Spoken Audio, Cambridge, U.K.*, pp. 117–122.

Hunter, J.: 2001, An overview of the MPEG-7 description definition language (DDL), *IEEE Transactions on Circuits and Systems for Video Technology* **11**(6), 765–772.

HyTime: 1997, Hypermedia, Time-based Structuring Language, *ISO/IEC JTC1/SC18/WG8 N1920 Second edition*, http://www.pms.ifi.lmu.de/mitarbeiter/ohlbach/multimedia/HYTIME/ISO/n1920.html.

Jaimes, A., Omura, K., Nagamine, T. and Hirata, K.: 2004, Memory cues for meeting video retrieval, *Proceedings of the the 1st ACM workshop on Continuous Archival and Retrieval of Personal Experiences, CARPE'04, New York, US*, ACM Press, New York, NY, US, pp. 74–85.

James, D. A. and Young, S. J.: 1994, A fast lattice-based approach to vocabulary independant worspotting, *Proceedings of International Conference on Acoustics, Speech, and Signal Processing, ICASSP-94, Adelaide, Australia*, Vol. 1, pp. 377–380.

Jeffries, R., Miller, J. R., Wharton, C. and Uyeda, K.: 1991, User interface evaluation in the real world: a comparison of four techniques, *Proceedings of the SIGCHI Conference on Human factors in Computing Systems, CHI '91, New Orleans, Louisiana, US*, ACM Press, New York, NY, US, pp. 119–124.

Jensen, C., Farnham, S. D., Drucker, S. M. and Kollock, P.: 2000, The effect of communication modality on cooperation in online environments, *Proceedings of the SIGCHI Conference on Human factors in Computing Systems, CHI '00, The Hague, Netherlands*, ACM Press, New York, NY, US, pp. 470–477.

Ju, W., Ionescu, A., Neeley, L. and Winograd, T.: 2004, Where the wild things work: capturing shared physical design workspaces, *Proceedings of the 2004 ACM Conference on Computer Supported Cooperative Work, CSCW '04, Chicago, US*, ACM Press, New York, NY, US, pp. 533–541.

Jurafsky, D. and Martin, J. H.: 2000, *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, Prentice Hall, Englewood Cliffs, New Jersey.

Koenen, R.: 1999, Mpeg-4: Multimedia for our time, *IEEE Spectrum* **36**(2), 26–33.

Koumpis, K. and Renals, S.: 2005, Content-based access to spoken audio, *IEEE Signal Processing Magazine* **22**(5), 61–69.

Lalanne, D., Ingold, R., von Rotz, D., Behera, A., Mekhaldi, D. and Popescu-Belis, A.: 2004, Using static documents as structured and thematic interfaces to multimedia meeting archives, *in* H. B. Samy Bengio (ed.), *Proceedings of First International Workshop on Machine Learning for Multimodal Interaction, MLMI 2004*, Vol. LNCS 3361/2005, Springer-Verlag GmbH, Martigny, Switzerland, pp. 291–304.

Lee, D.-S., Erol, B., Graham, J., Hull, J. J. and Murata, N.: 2002, Portable meeting recorder, *Proceedings of the Tenth ACM International Conference on Multimedia, MULTIMEDIA '02, Juan-les-Pins, France*, ACM Press, New York, NY, US, pp. 493–502.

Lee, D.-S., Hull, J., Erol, B. and Graham, J.: 2004, MinuteAid: multimedia note-taking in an intelligent meeting room, *IEEE International Conference on Multimedia and Expo, Baltimore, MD, US*, Vol. 3, IEEE press, pp. 1759 – 1762.

Li, F. C., Gupta, A., Sanocki, E., wei He, L. and Rui, Y.: 2000, Browsing digital video, *Proceedings of the SIGCHI Conference on Human factors in Computing Systems, CHI '00, The Hague, Netherlands*, ACM Press, New York, NY, US, pp. 169–176.

Liou, M.: 1991, Overview of the p
x64 kbit/s video coding standard, *Communications of the ACM* **34**(4), 59–63.

Little, T. D. and Ghafoor, A.: 1990, Multimedia objects models for synchronization and databases, *Proceedings of Sixth International Conference on Data Engineering*, Los Angeles, CA, pp. 20 – 27.

Luz, S.: 2002, Interleave factor and multimedia information visualisation, *in* H. Sharp, P. Chalk, J. LePeuple and J. Rosbottom (eds), *Proceedings of Human Computer Interaction 2002*, Vol. 2, London, pp. 142–146.

Luz, S. and Bouamrane, M.-M.: 2006, Exploring the structure of media stream interactions for multimedia browsing, *in* M. Detyniecki, J. M. Jose, A. Nrnberger and C. J. Rijsbergen (eds), *Proceedings of Adaptive Multimedia Retrieval: User, Context, and Feedback: Third International Workshop, AMR 2005, Revised Selected Papers,*, Vol. LNCS 3877 / 2006, Glasgow, pp. 79–90.

Luz, S., Bouamrane, M.-M. and Masoodian, M.: 2006, Gathering a corpus of multimodal computer-mediated meetings with focus on text and audio interaction, *Proceedings of the fifth International Conference on Language Resources and Evaluation, LREC06*, Genoa, Italy, pp. 407–412.

Luz, S. and Masoodian, M.: 2004, A mobile system for non-linear access to time-based data, *Proceedings of the Conference on Advanced Visual Interfaces, AVI '04, Gallipoli, Italy*, ACM Press, New York, NY, US, pp. 454–457.

Luz, S. and Masoodian, M.: 2005, A model for meeting content storage and retrieval, *Proceedings of the 11th International Multimedia Modelling Conference, MMM'05*, IEEE Press, pp. 392–398.

Luz, S. and Roy, D.: 1999, Meeting browser: A system for visualising and accessing audio in multicast meetings., *Proceedings of the International Workshop on Multimedia Signal Processing*, IEEE Signal Processing Society.

Martinez, J.: 2002, Standards - MPEG-7 overview of MPEG-7 description tools, part 2, *IEEE Multimedia* **9**(3), 83–93.

Martinez, J., Koenen, R. and Pereira, F.: 2002, Mpeg-7: the generic multimedia content description standard, part 1, *IEEE Multimedia* **9**(2), 78–87.

Masoodian, M., Luz, S., Bouamrane, M.-M. and King, D.: 2005, RECOLED: a group-aware collaborative text editor for capturing document history, *Proceedings of WWW/Internet 2005*, Vol. 1, Lisbon, pp. 323–330.

McCowan, I., Gatica-Perez, D., Bengio, S., Lathoud, G., Barnard, M. and Zhang, D.: 2005, Automatic analysis of multimodal group actions in meetings, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* **27**(3), 305–317.

McLellan, P., Tombros, A., Jose, J. M., Ounis, I. and Whitehead, M.: 2001, Evaluating Summarisation Technologies: A Task Oriented Approach, *Proceedings of the 1st International Workshop on New Developments in Digital Libraries, NDDL '01*, ICEIS Press, pp. 99–112.

Meyer-Boudnik, T. and Effelsberg, W.: 1995, MHEG Explained, *IEEE Multimedia* **2**(1), 26–38.

MHEG: 1991, Multimedia and Hypermedia information coding Expert Group, *ISO/IEC JTC1/SC29/WG12 (MHEG), http://www.mheg.org.*

Moran, T. P., Palen, L., Harrison, S., Chiu, P., Kimber, D., Minneman, S., van Melle, W. and Zellweger, P.: 1997, "I'll get that off the audio": A case study of salvaging multimedia meeting records, *Proceedings of ACM Conference on Human factors in Computing Systems, CHI 97, Atlanta, US*, Vol. 1, pp. 202–209.

MPEG: 1988, Moving Picture Experts Group, *ISO/IEC JTC 1/SC 29/ WG 11 (MPEG)*, http://www.mpeg.org/MPEG/index.html.

MPEG-4: 2002, Moving Picture Experts Group - 4, *ISO/IEC JTC1/SC29/ WG 11 (MPEG)*, http://www.chiariglione.org/mpeg/standards/mpeg-4/mpeg-4.htm.

MPEG-7: 2004, Moving Picture Experts Group - 7, *ISO/IEC JTC 1/SC 29/ WG11 (MPEG)*, http://www.chiariglione.org/MPEG/standards/mpeg-7/mpeg-7.htm.

Nakatani, C., Whittaker, S. and Hirschberg, J.: 1998, Now you hear it, now you don't: Empirical studies of audio browsing behavior, *Proceedings of International Conference on Spoken Language Processing, ICSLP 1998*, Vol. 4, Sydney, Australia, pp. 1651–1654.

Nielsen, J. and Molich, R.: 1990, Heuristic evaluation of user interfaces, *CHI '90: Proceedings of the SIGCHI Conference on Human factors in Computing Systems, Seattle, Washington, US*, ACM Press, New York, NY, US, pp. 249–256.

Peterson, J. L.: 1977, Petri nets, *ACM Computing Surveys* **9**(3), 223–252.

Pfeiffer, S.: 2001, Pause concepts for audio segmentation at different semantic levels, *Proceedings of the Ninth ACM International Conference on Multimedia, MULTIMEDIA '01, Ottawa, Ontario, Canada*, ACM Press, New York, NY, US, pp. 187–193.

Pickering, M. J. and Branigan, H. P.: 1999, Syntactic priming in language production, *Trends in Cognitive Sciences* **3**(4), 136–141.

Porter, M.: 1980, An algorithm for suffix stripping, *Program* **14**(3), 130–137.

Posner, I. and Baecker: 1992, How people write together, *Proceedings of the Twenty-fifth Annual Hawaii International Conference on System Sciences, Hawaii, US*, Morgan Kaufmann, pp. 127–138.

Preece, J., Rogers, Y. and Sharp, H.: 2002, *Interaction Design: Beyond Human-Computer Interaction*, John Wiley and sons.

Rabiner, L. R. and Juang, B.-H.: 1993, *Fundamentals of Speech Recognition*, Prentice Hall.

Rector, A.: 1999, Clinical terminology: Why is it so hard?, *Methods of Information in Medicine* **38**, 239 – 252.

Richter, H. A., Abowd, G. D., Geyer, W., Fuchs, L., Daijavad, S. and Poltrock, S. E.: 2001, Integrating meeting capture within a collaborative team environment, *Proceedings of the 3rd International Conference on Ubiquitous Computing, UbiComp '01, Atlanta, GA, US*, Springer-Verlag, London, UK, pp. 123–138.

Rohlicek, J., Russell, W., Roukos, S. and Gish, H.: 1989, Continuous hidden Markov modeling for speaker-independent word spotting, *Proceedings of International Conference on Acoustics, Speech and Signal Processing, ICASSP-89, Glasgow, Scotland*, Vol. 1, pp. 627–630.

Rose, R. C. and Paul, D. B.: 1990, A hidden Markov model based keyword recognition system, *Proceedings of International Conference on Acoustics, Speech, and Signal Processing, ICASSP-90*, Vol. 1, Albuquerque, US, pp. 129–132.

Roy, D. and Malamud, C.: 1997, Speaker identification based text to audio alignment for an audio retrieval system, *Proceedings of the 1997 IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP '97, Muenchen, Germany*, Vol. 2, IEEE Computer Society, Washington, DC, US, pp. 1099–1102.

Russell, D. M.: 2000, A design pattern-based video summarization technique: Moving from low-level signals to high-level structure, *Proceedings of the 33rd Hawaii International Conference on System Sciences, HICSS '00, Hawaii, US*, Vol. 3, IEEE Computer Society, Washington, DC, USA, p. 3048.

Rutledge, L.: 2001, SMIL 2.0: XML for Web multimedia, *IEEE Journal on Internet Computing* **5**(5), 78–84.

Rutledge, L., Hardman, L., Ossenbruggen, J. V. and Bulterman, D.: 1999, Adaptable hypermedia with web standards and tools, *Proceedings of The Active Web 1999, Staffordshire, U.K.*, http://www.visualize.uk.com/conf/activeweb/proceed/contents.asp.

Salembier, P. and Smith, J.: 2001, MPEG-7 multimedia description schemes, *IEEE Transactions on Circuits and Systems for Video Technology* **11**(6), 748–759.

Shriberg, E., Stolcke, A., Hakkani-Tur, D. and Tur, G.: 2000, Prosody-based automatic segmentation of speech into sentences and topics, *Speech Communications,* **32**(1-2), 127–154.

Smeaton, A. F.: 2001, Indexing, browsing, and searching of digital video and digital audio information, *Third European Summer School on Information Retrieval, (ESSIR2000) Varenna, Italy, LNCS Lectures on Information Retrieval* pp. 93–110.

SMIL: 2001, Synchronized Multimedia Integration Language, *[Second Edition]*, http://www.w3.org/TR/SMIL20/.

Smith, M. A. and Kanade, T.: 1998, Video skimming and characterization through the combination of image and language understanding techniques, *Proceedings of IEEE International Workshop on Content-Based Access of Image and Video Database, Bombay, India*, IEEE Computer Society, pp. 61–70.

Smith, M. and Wheeldon, L.: 2001, Syntactic priming in spoken sentence production - an online study, *Cognition* **78**, 123–164.

Snoek, C. G. M. and Worring, M.: 2005, Multimodal video indexing: A review of the state-of-the-art, *Multimedia Tools and Applications* **25**(1), 5–35.

Srinivasan, S., Ponceleon, D., Amir, A. and Petkovic, D.: 1999, What is in that video anyway?: in search of better browsing, *Proceedings of IEEE International Conference on Multimedia Computing and Systems*, Vol. 1, Florence, Italy, pp. 388–393.

Stifelman, L., Arons, B. and Schmandt, C.: 2001, The Audio Notebook: paper and pen interaction with structured speech, *Proceedings of the SIGCHI Conference on Human factors in Computing Systems, CHI '01, Seattle, WA, US*, ACM Press, New York, NY, US, pp. 182–189.

Sun, C. and Chen, D.: 2002, Consistency maintenance in real-time collaborative graphics editing systems, *ACM Transactions on Computer-Human Interaction* **9**(1), 1–41.

Sun, C., Jia, X., Yang, Y. and Zhang, Y.: 1997, REDUCE: A Prototypical Cooperative Editing System, *Proceedings of the Seventh International Conference on Human-Computer Interaction, HCI International '97, San Francisco, US*, Vol. 1, Elsevier Science Inc., New York, NY, US, pp. 89–92.

Sun, C., Jia, X., Zhang, Y., Yang, Y. and Chen, D.: 1998, Achieving convergence, causality preservation, and intention preservation in real-time cooperative editing systems, *ACM Transactions on Computer-Human Interaction,* **5**(1), 63–108.

Tannen, D.: 1989, *Talking Voices, Repetition, Dialogue and Imagery in Conversational Discourse*, Studies in interactional sociolinguistics, Cambridge University Press.

Tucker, S. and Whittaker, S.: 2005, Accessing multimodal meeting data: Systems, problems and possibilities, *in* H. B. Samy Bengio (ed.), *Machine Learning for Multimodal Interaction: First International Workshop, MLMI 2004, Martigny, Switzerland*, Vol. 3361, Springer-Verlag GmbH, pp. 1–11.

Tur, G., Hakkani-Tur, D., Stolcke, A. and Shriberg, E.: 2001, Integrating prosodic and lexical cues for automatic topic segmentation, *Computer Linguistics,* **27**(1), 31–57.

Uchihashi, S., Foote, J., Girgensohn, A. and Boreczky, J.: 1999, Video Manga: generating semantically meaningful video summaries, *Proceedings of the Seventh ACM International Conference on Multimedia, MULTIMEDIA '99, Orlando, FL, US*, ACM Press, New York, NY, US, pp. 383–392.

Valenza, R., Robinson, T., Hickey, M. and Tucker, R.: 1999, Summarisation of spoken audio through information extraction, *Proceedings of the ESCA workshop: Accessing information in Spoken Audio*, Cambridge, UK, pp. 111–115.

van Leeuwen, J.: 2003, Computer support for collaborative work in the construction industry, *Proceedings of the International Conference on Concurrent Engineering* pp. 599–606.

van Rijsbergen, C.: 1979, *Information Retrieval*, Butterworths, London, UK, http://www.dcs.gla.ac.uk/ iain/keith/index.htm.

Wahl, T. and Rothermel, K.: 1994, Representing time in multimedia systems, *International Conference on Multimedia Computing and Systems, Boston, Massachusetts, US*, IEEE Press, pp. 538–543.

Waibel, A., Bett, M., Finke, M. and Stiefelhagen, R.: 1998, Meeting browser: Tracking and summarizing meetings, *in* D. E. M. Penrose (ed.), *Proceedings of the Broadcast News Transcription and Understanding Workshop, Lansdowne, Virginia, US*, Morgan Kaufmann, pp. 281–286.

Waibel, A., Bett, M., Metze, F., Ries, K., Schaaf, T., Schultz, T., Soltau, H., Yu, H. and Zechner, K.: 2001, Advances in automatic meeting record creation and access, *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, pp. 597–600.

Walker, W., P. Lamere, P. K., Raj, B., Singh, R., Gouvea, E., Wolf, P. and Woelfel, J.: 2004, Sphinx-4: A flexible open source framework for speech recognition, *Sun Microsystems, Technical Report TR-2004-139* .

Weintraub, M.: 1993, Keyword-spotting using SRI's DECIPHER large-vocabularyspeech-recognition system, *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP-93, Minneapolis, US*, Vol. 2, pp. 463–466.

Wellner, P., Flynn, M. and Guillemot, M.: 2004, Browsing recorded meetings with Ferret, *in* S. Bengio and H. Bourlard (eds), *Proceedings of Machine Learning for Multimodal Interaction: First International Workshop, MLMI 2004, Martigny, Switzerland*, Vol. 3361, Springer-Verlag GmbH, pp. 12–21.

Wellner, P., Flynn, M., Tucker, S. and Whittaker, S.: 2005, A meeting browser evaluation test, *Extended abstracts on Human factors in Computing Systems, CHI '05, Oregon, US*, ACM Press, New York, NY, US, pp. 2021–2024.

Whittaker, S., Hirschberg, J., Choi, J., Hindle, D., Pereira, F. and Singhal, A.: 1999, SCAN: designing and evaluating user interfaces to support retrieval from speech archives, *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '99, Berkeley, CA, US*, ACM Press, New York, NY, US, pp. 26–33.

Whittaker, S., Hirschberg, J. and Nakatani, C. H.: 1998a, All talk and all action: strategies for managing voicemail messages, *Conference Summary on Human factors in Computing Systems, CHI 98, Los Angeles, CA, US*, ACM Press, New York, NY, US, pp. 249–250.

Whittaker, S., Hirschberg, J. and Nakatani, C. H.: 1998b, Play it again: a study of the factors underlying speech browsing behavior, *Proceedings of Conference on Human factors in Computing Systems, CHI '98*, ACM Press, LA, US, pp. 247–248.

Whittaker, S., Hyland, P. and Wiley, M.: 1994, FILOCHAT: handwritten notes provide access to recorded conversations., *Proceedings of the ACM Conference on Human factors in Computing Systems, CHI'94, Boston, US*, ACM Press, New York, US, pp. 24–28.

Wilcox, L., Kimber, D. and Chen, F.: 1994, Audio indexing using speaker identification, *Proceedings of Conference on Automatic Systems for the Inspection and Identification of Humans, San Diego, California*, pp. 149–157.

XMLSchema: 2000, eXtensible Markup Language Schema, *W3C*, http://www.w3.org/XML/Schema.

Yamron, J., Carp, I., Gillick, L., Lowe, S. and van Mulbregt, P.: 1997, Event tracking and text segmentation via hidden Markov models, *Proceedings of IEEE Workshop on Automatic Speech Recognition and Understanding*, Santa Barbara, CA, US, pp. 519–526.

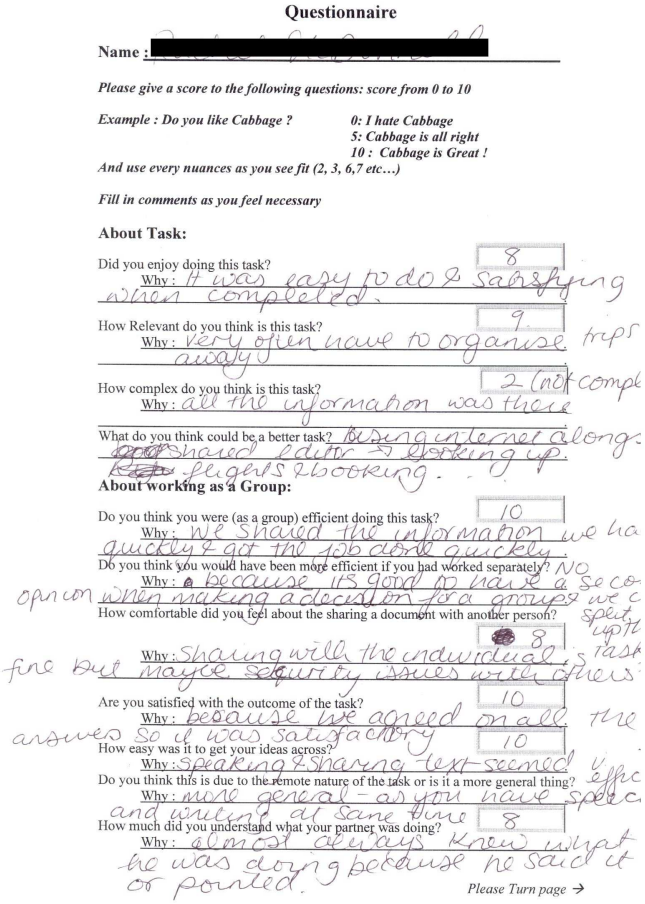
Young, S.: 1995, Large vocabulary continuous speech recognition: A review, *Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding*, Snowbird, Utah, US, pp. 3–28.

Zechner, K.: 2001, Automatic generation of concise summaries of spoken dialogues in unrestricted domains, *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval,SIGIR '01*, ACM Press, New York, NY, US, pp. 199–207.

Zechner, K. and Waibel, A.: 2000, DiaSumm: flexible summarization of spontaneous dialogues in unrestricted domains, *Proceedings of the 18th Conference on Computational Linguistics, Morristown, NJ, US*, Association for Computational Linguistics, pp. 968–974.

Appendix A

RECOLED Usability Questionnaire



Questionnaire

Name : [redacted]

Please give a score to the following questions: score from 0 to 10

Example : Do you like Cabbage ? ***0: I hate Cabbage***
5: Cabbage is all right
10 : Cabbage is Great !

And use every nuances as you see fit (2, 3, 6,7 etc…)

Fill in comments as you feel necessary

About Task:

Did you enjoy doing this task? 8
Why : It was easy to do & satisfying when completed.

How Relevant do you think is this task? 9
Why : very often have to organise trips away

How complex do you think is this task? 2 (not compl
Why : all the information was there

What do you think could be a better task? Using internet along. shared editor to looking up. flights & booking.

About working as a Group:

Do you think you were (as a group) efficient doing this task? 10
Why : We shared the information we ha quickly & got the job done quickly.

Do you think you would have been more efficient if you had worked separately? NO
Why : because it's good to have a se co opinion when making a decision for a groups we c

How comfortable did you feel about the sharing a document with another person? split up th
8
Why : Sharing with the individual s task fine but maybe security issues with others

Are you satisfied with the outcome of the task? 10
Why : because we agreed on all the answers so it was satisfactory

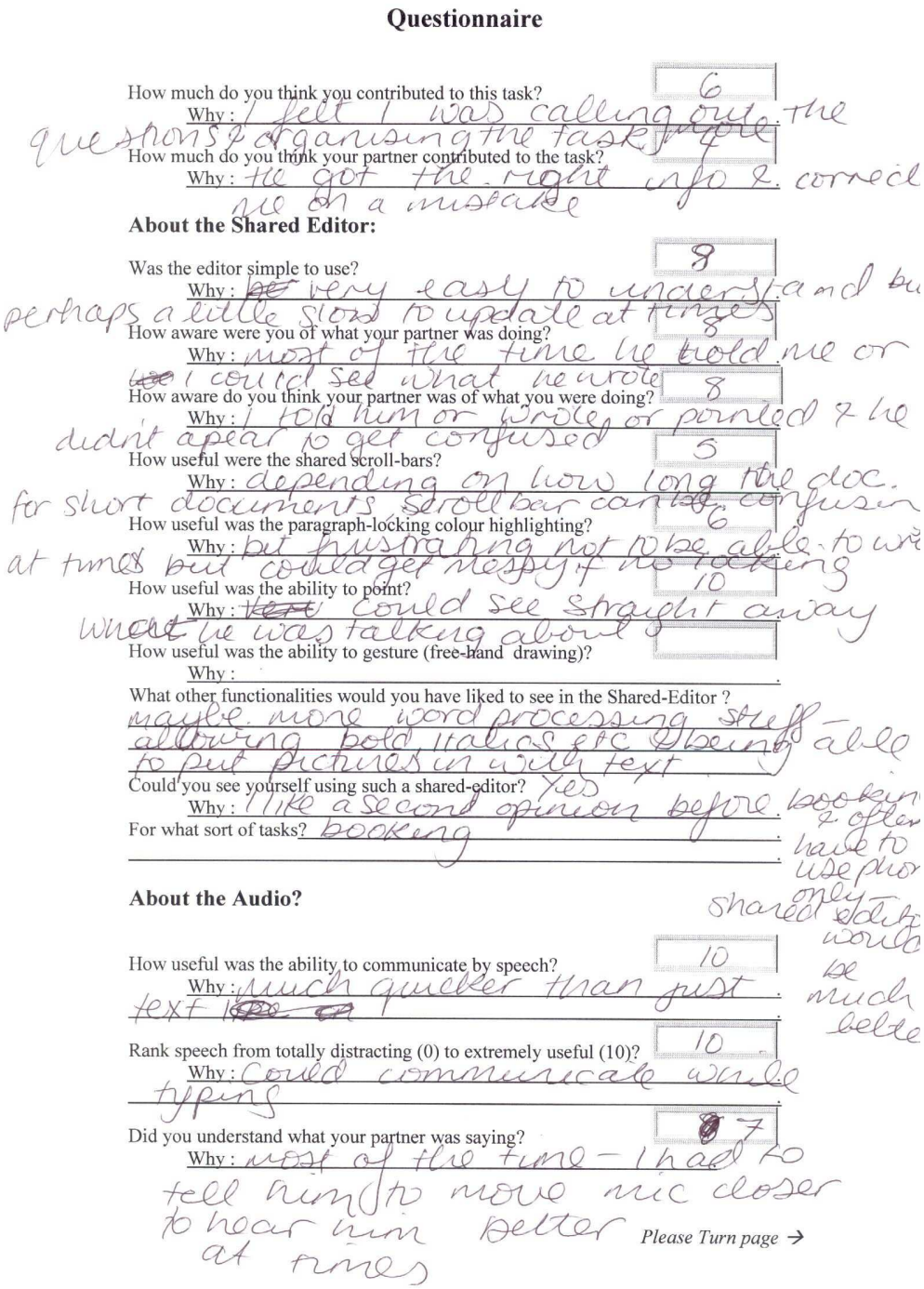
How easy was it to get your ideas across? 10
Why : Speaking & sharing text seemed v effe

Do you think this is due to the remote nature of the task or is it a more general thing?
Why : more general - as you have spea and writing at same time

How much did you understand what your partner was doing? 8
Why : almost always knew what he was doing because he said it or pointed.

Please Turn page →

Figure A.1: RECOLED Usability Questionnaire: page 1 of 2



Questionnaire

How much do you think you contributed to this task? 6
Why : I felt I was calling out the questions & organising the task
How much do you think your partner contributed to the task? 4
Why : He got the right info & corrected me on a mistake

About the Shared Editor:

Was the editor simple to use? 8
Why : very easy to understand but perhaps a little slow to update at times
How aware were you of what your partner was doing? 8
Why : most of the time he told me or I could see what he wrote
How aware do you think your partner was of what you were doing? 8
Why : I told him or wrote or pointed & he didn't apear to get confused
How useful were the shared scroll-bars? 5
Why : depending on how long the doc. for short documents, scroll bar can be confusing
How useful was the paragraph-locking colour highlighting? 6
Why : but frustrating not to be able to write at times but could get messy if no locking
How useful was the ability to point? 10
Why : could see straight away what he was talking about
How useful was the ability to gesture (free-hand drawing)?
Why :
What other functionalities would you have liked to see in the Shared-Editor ?
maybe more word processing stuff allowing bold, italics etc & being able to put pictures in with text
Could you see yourself using such a shared-editor? Yes
Why : like a second opinion before booking & often have to use phone only shared editor would be much better
For what sort of tasks? booking

About the Audio?

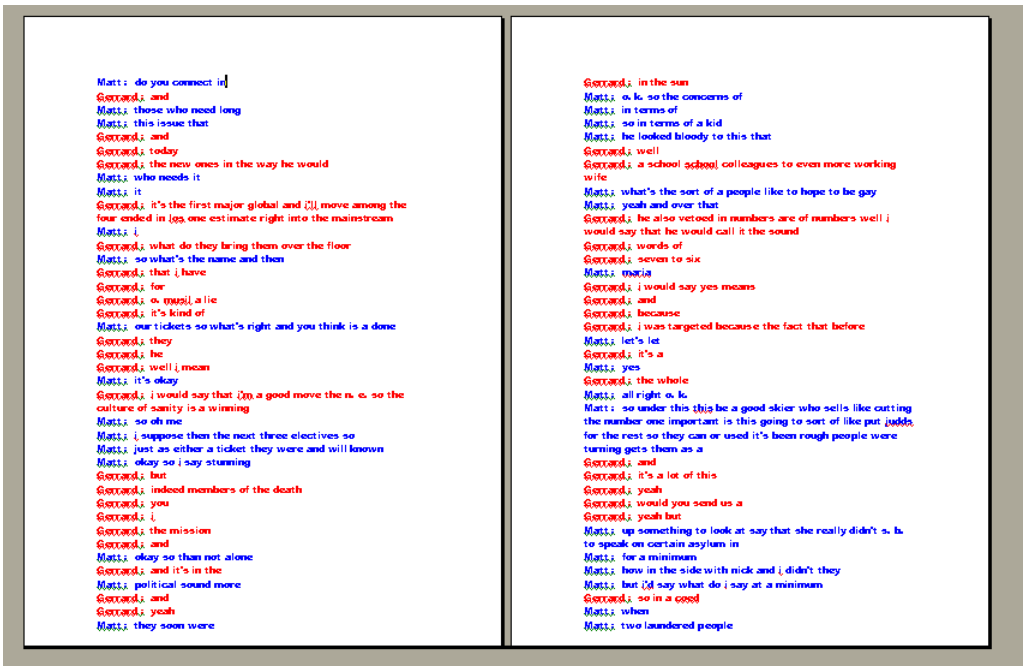
How useful was the ability to communicate by speech? 10
Why : much quicker than just text
Rank speech from totally distracting (0) to extremely useful (10)? 10
Why : Could communicate while typing
Did you understand what your partner was saying? 7
Why : most of the time - I had to tell him to move mic closer to hear him better at times

Please Turn page →

Figure A.2: RECOLED Usability Questionnaire: page 2 of 2

Appendix B

ASR Speech Transcripts of a Meeting



Matt : do you connect in
Gerrard : and
Matt : those who need long
Matt : this issue that
Gerrard : and
Gerrard : today
Gerrard : the new ones in the way he would
Matt : who needs it
Matt : it
Gerrard : it's the first major global and i'll move among the four ended in ios one estimate right into the mainstream
Matt : i
Gerrard : what do they bring them over the floor
Matt : so what's the name and then
Gerrard : that i have
Gerrard : for
Gerrard : o. mysil a lie
Gerrard : it's kind of
Matt : our tickets so what's right and you think is a done
Gerrard : they
Gerrard : he
Gerrard : well i mean
Matt : it's okay
Gerrard : i would say that i'm a good move the n. e. so the culture of sanity is a winning
Matt : so oh me
Matt : i suppose then the next three electives so
Matt : just as either a ticket they were and will known
Matt : okay so i say stunning
Gerrard : but
Gerrard : indeed members of the death
Gerrard : you
Gerrard : i
Gerrard : the mission
Gerrard : and
Matt : okay so than not alone
Gerrard : and it's in the
Matt : political sound more
Gerrard : and
Gerrard : yeah
Matt : they soon were

Gerrard : in the sun
Matt : o. k. so the concerns of
Matt : in terms of
Matt : so in terms of a kid
Matt : he looked bloody to this that
Gerrard : well
Gerrard : a school school colleagues to even more working wife
Matt : what's the sort of a people like to hope to be gay
Matt : yeah and over that
Gerrard : he also vetoed in numbers are of numbers well i would say that he would call it the sound
Gerrard : words of
Gerrard : seven to six
Matt : macia
Gerrard : i would say yes means
Gerrard : and
Gerrard : because
Gerrard : i was targeted because the fact that before
Matt : let's let
Gerrard : it's a
Matt : yes
Gerrard : the whole
Matt : all right o. k.
Matt : so under this this be a good skier who sells like cutting the number one important is this going to sort of like put judds for the rest so they can or used it's been rough people were turning gets them as a
Gerrard : and
Gerrard : it's a lot of this
Gerrard : yeah
Gerrard : would you send us a
Gerrard : yeah but
Matt : up something to look at say that she really didn't s. h. to speak on certain asylum in
Matt : for a minimum
Matt : how in the side with nick and i didn't they
Matt : but i'd say what do i say at a minimum
Gerrard : so in a coed
Matt : when
Matt : two laundered people

Figure B.1: ASR Speech transcripts of 1 hour long meeting with 2 participants: first 2 pages

Figure B.2: ASR Speech transcripts of 1 hour long meeting with 2 participants: page 1 to 12

Figure B.3: ASR Speech transcripts of 1 hour long meeting with 2 participants: page 13 of 23

Appendix C

Results of Analytic Evaluation of Topic Search

Table C.1: Topic Search results according to the Task-Oriented users' queries on meeting A

Topic Search	True Hit	False Alarm	Miss	No Match
"Advertise" + "Date"		1		
"Advertise" + "Radio"	1		1	
"Advertise" + "Week"	1			
"Band" + "Euro" + "Ticket"				X
"Band" + "Support"	3		3	
"Band" + "Ticket"			2	
"Capacity" + "Friday"				X
"College" + "Friday"				X
"Date" + "Night"	1		1	
"Friday" + "People"	1		1	
"Friday" + "Night" + "Thursday"	4		2	
"Friday" + "Thursday"	4		2	
"Gig" + "Night"	1		1	
"Losers" + "Support"			1	
"Olympia" + "TBMC"	1			
"Polar" + "Thursday"			2	
"Price" + "Quid"	1		2	
"Price" + "Ticket"	3		2	
"Venue" + "Village"	2			
"Venue" + "Village" + "Whelans"	2			
"Village" + "Whelans"	1			

Table C.2: Topic Search results according to the Task-Oriented users' queries on meeting B

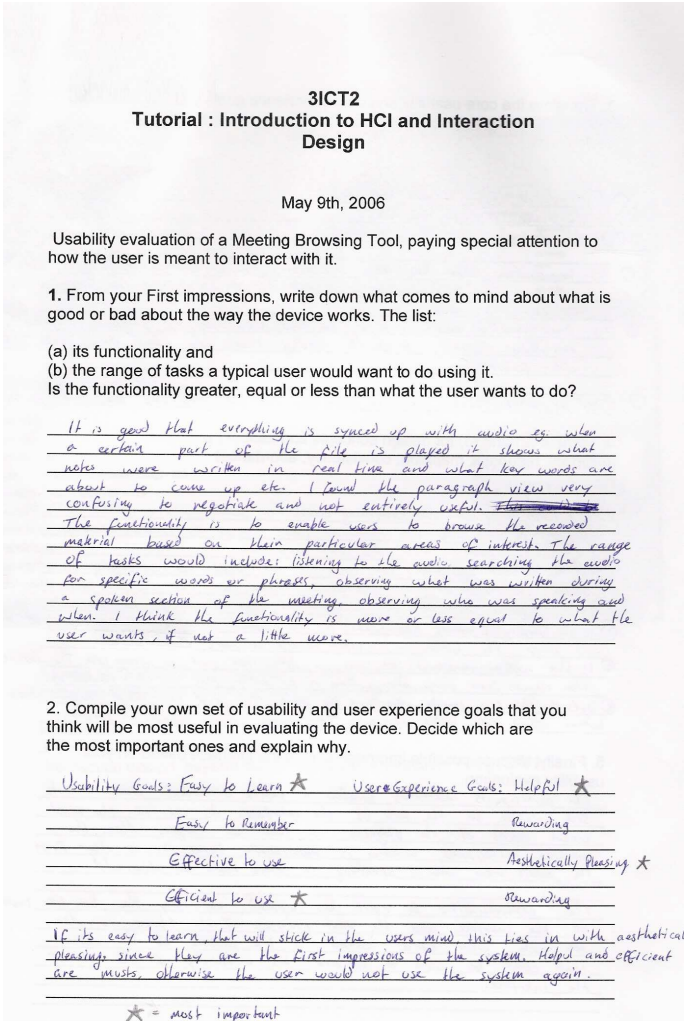
Topic Search	True Hit	False Alarm	Miss	No Match
"Beijing" + "Bus" + "Wall"	2			
"Beijing" + "Flight" + "Internal"	1	1		
"Beijing" + "Hong Kong"			1	
"Beijing" + "Hong Kong" + "Start"			1	
"Beijing" + "Quan"				X
"Buddhist" + "Shui"				X
"Bus" + "Great"	3	1	2	
"Bus" + "Great" + "Wall"	3	1	2	
"Bus" + "Wall"	3	1	2	
"Cable" + "Car"	2			
"Cable" + "Car" + "Pony"	2			
"Day" + "Ping Yao"			3	
"Day" + "Shangai"		1	1	
"Day" + "Extra" + "Ping Yao"	1			
"Duck" + "Restaurant"	1			
"Famous" + "Quan"	1			
"Famous" + "Restaurant"	1			
"Forbidden" + "City"	1		4	
"Forbidden" + "Quan"			1	
"Great" + "Wall"	4		2	
"Guilin" + "Million"			1	
"Guilin" + "Train"	1		2	
"Lunan" + "Market"	1			
"Night" + "Train"	6		3	
"Ping Yao" + "Town"	1			
"People" + "Terracotta" + "Xian"		1		
"Quan" + "Ju" + "De"	1			

Table C.3: Topic Search results aummary on meetings A & B

Topic Search	True Hit	False Alarm	Miss	No Match
TOTAL	61	7	45	5
Percentage of Total	51.7%	5.9%	38.1%	4.2%

Appendix D

Heuristic Evaluation Report



3ICT2
Tutorial : Introduction to HCI and Interaction Design

May 9th, 2006

Usability evaluation of a Meeting Browsing Tool, paying special attention to how the user is meant to interact with it.

1. From your First impressions, write down what comes to mind about what is good or bad about the way the device works. The list:

(a) its functionality and
(b) the range of tasks a typical user would want to do using it.
Is the functionality greater, equal or less than what the user wants to do?

It is good that everything is synced up with audio eg. when a certain part of the file is played it shows what notes were written in real time and what key words are about to come up etc. I found the paragraph view very confusing to negotiate and not entirely useful. ~~The~~
The functionality is to enable users to browse the recorded material based on their particular areas of interests. The range of tasks would include: listening to the audio, searching the audio for specific words or phrases, observing what was written during a spoken section of the meeting, observing who was speaking and when. I think the functionality is more or less equal to what the user wants, if not a little more.

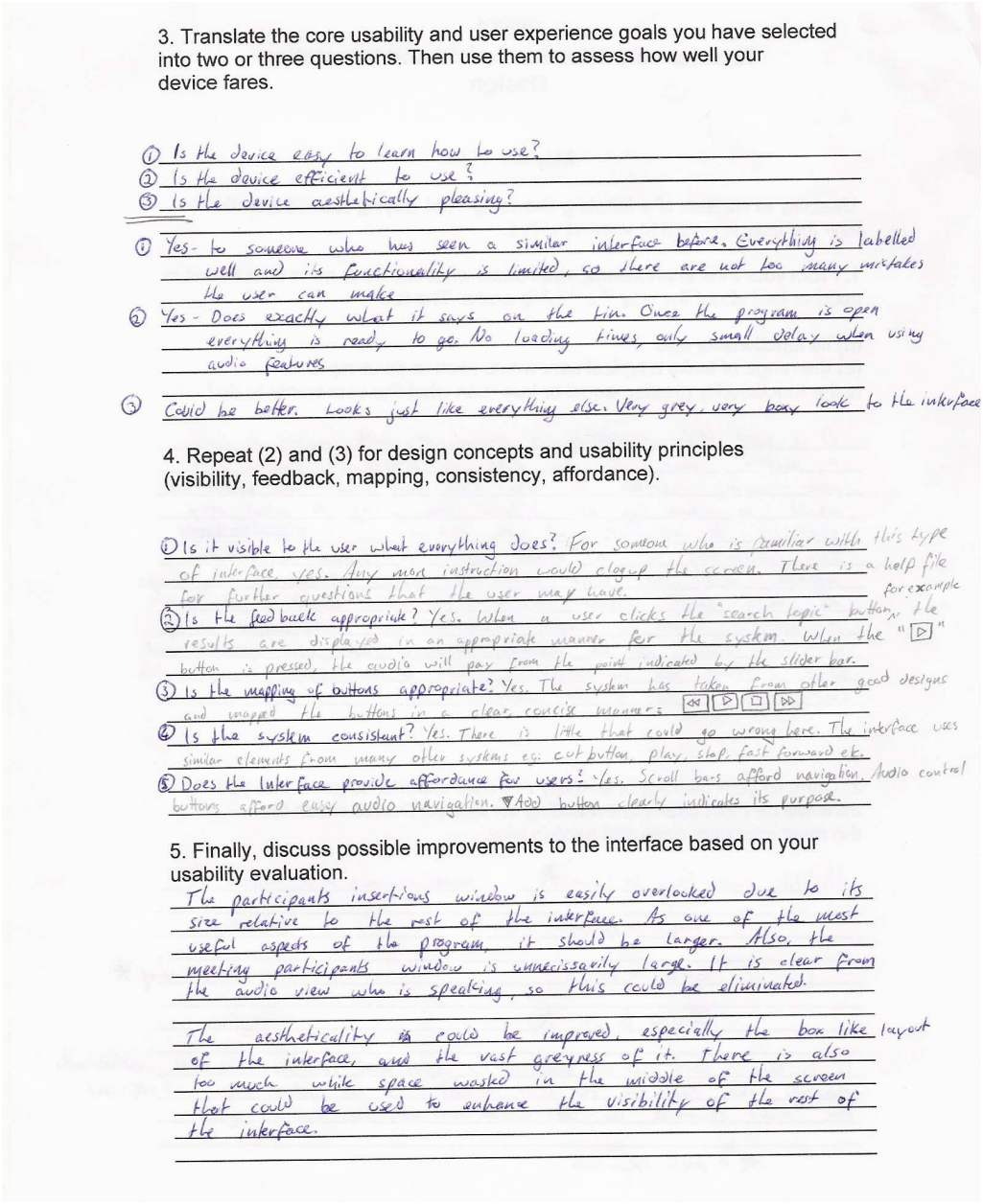
2. Compile your own set of usability and user experience goals that you think will be most useful in evaluating the device. Decide which are the most important ones and explain why.

Usability Goals:	User Experience Goals:
Easy to Learn ★	Helpful ★
Easy to Remember	Rewarding
Effective to use	Aesthetically Pleasing ★
Efficient to use ★	Satisfying

If its easy to learn, that will stick in the users mind, this ties in with aesthetical pleasing since they are the first impressions of the system. Helpful and efficient are musts, otherwise the user would not use the system again.

★ = most important

Figure D.1: Form used for Heuristic Evaluation: page 1 of 2



3. Translate the core usability and user experience goals you have selected into two or three questions. Then use them to assess how well your device fares.

① Is the device easy to learn how to use?
② Is the device efficient to use?
③ Is the device aesthetically pleasing?

① Yes - to someone who has seen a similar interface before. Everything is labelled well and its functionality is limited, so there are not too many mistakes the user can make

② Yes - Does exactly what it says on the tin. Once the program is open everything is ready to go. No loading times, only small delay when using audio features

③ Could be better. Looks just like everything else. Very grey, very boxy look to the interface

4. Repeat (2) and (3) for design concepts and usability principles (visibility, feedback, mapping, consistency, affordance).

① Is it visible to the user what everything does? For someone who is familiar with this type of interface, yes. Any more instruction would clog up the screen. There is a help file for further questions that the user may have.

② Is the feedback appropriate? Yes. When a user clicks the "search topic" button, for example the results are displayed in an appropriate manner for the system. When the "▷" button is pressed, the audio will play from the point indicated by the slider bar.

③ Is the mapping of buttons appropriate? Yes. The system has taken from other good designs and mapped the buttons in a clear, concise manner. ⏮ ▷ □ ⏭

④ Is the system consistent? Yes. There is little that could go wrong here. The interface uses similar elements from many other systems eg: cut button, play, stop, fast forward etc.

⑤ Does the interface provide affordance for users? Yes. Scroll bars afford navigation. Audio control buttons afford easy audio navigation. ▼Add button clearly indicates its purpose.

5. Finally, discuss possible improvements to the interface based on your usability evaluation.

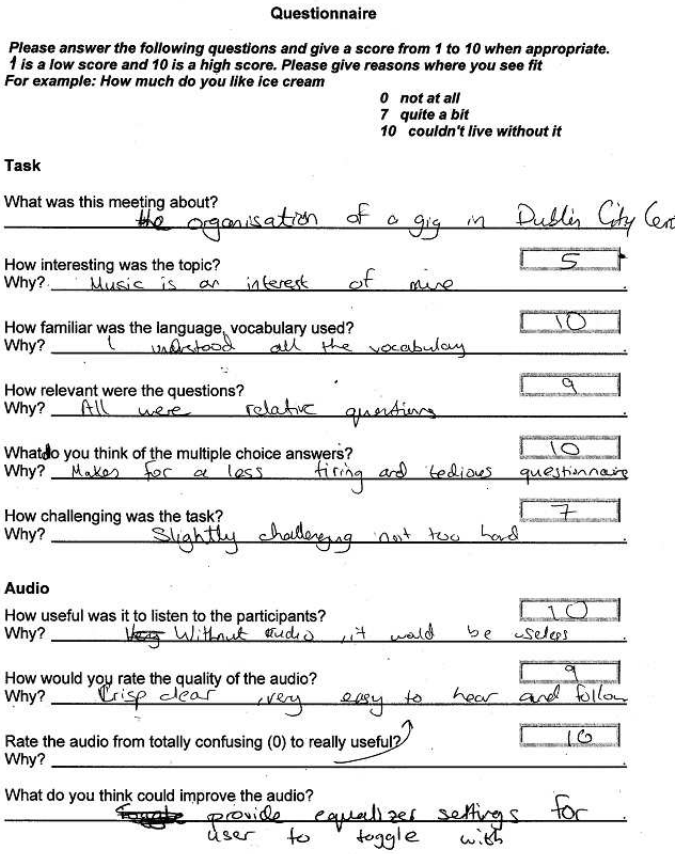
The participants insertions window is easily overlooked due to its size relative to the rest of the interface. As one of the most useful aspects of the program, it should be larger. Also, the meeting participants window is unnecessarily large. It is clear from the audio view who is speaking, so this could be eliminated.

The aesthetically could be improved, especially the box like layout of the interface, and the vast greyness of it. There is also too much white space wasted in the middle of the screen that could be used to enhance the visibility of the rest of the interface.

Figure D.2: Form used for Heuristic Evaluation: page 2 of 2

Appendix E

MeetingMiner Usability Questionnaire



Questionnaire

Please answer the following questions and give a score from 1 to 10 when appropriate. 1 is a low score and 10 is a high score. Please give reasons where you see fit
For example: How much do you like ice cream

0 not at all
7 quite a bit
10 couldn't live without it

Task

What was this meeting about?
~~the~~ organisation of a gig in Dublin City Cen

How interesting was the topic? 5
Why? Music is an interest of mine

How familiar was the language, vocabulary used? 10
Why? I understood all the vocabulary

How relevant were the questions? 9
Why? All were relative questions

What do you think of the multiple choice answers? 10
Why? Makes for a less tiring and tedious questionnaire

How challenging was the task? 7
Why? Slightly challenging not too hard

Audio

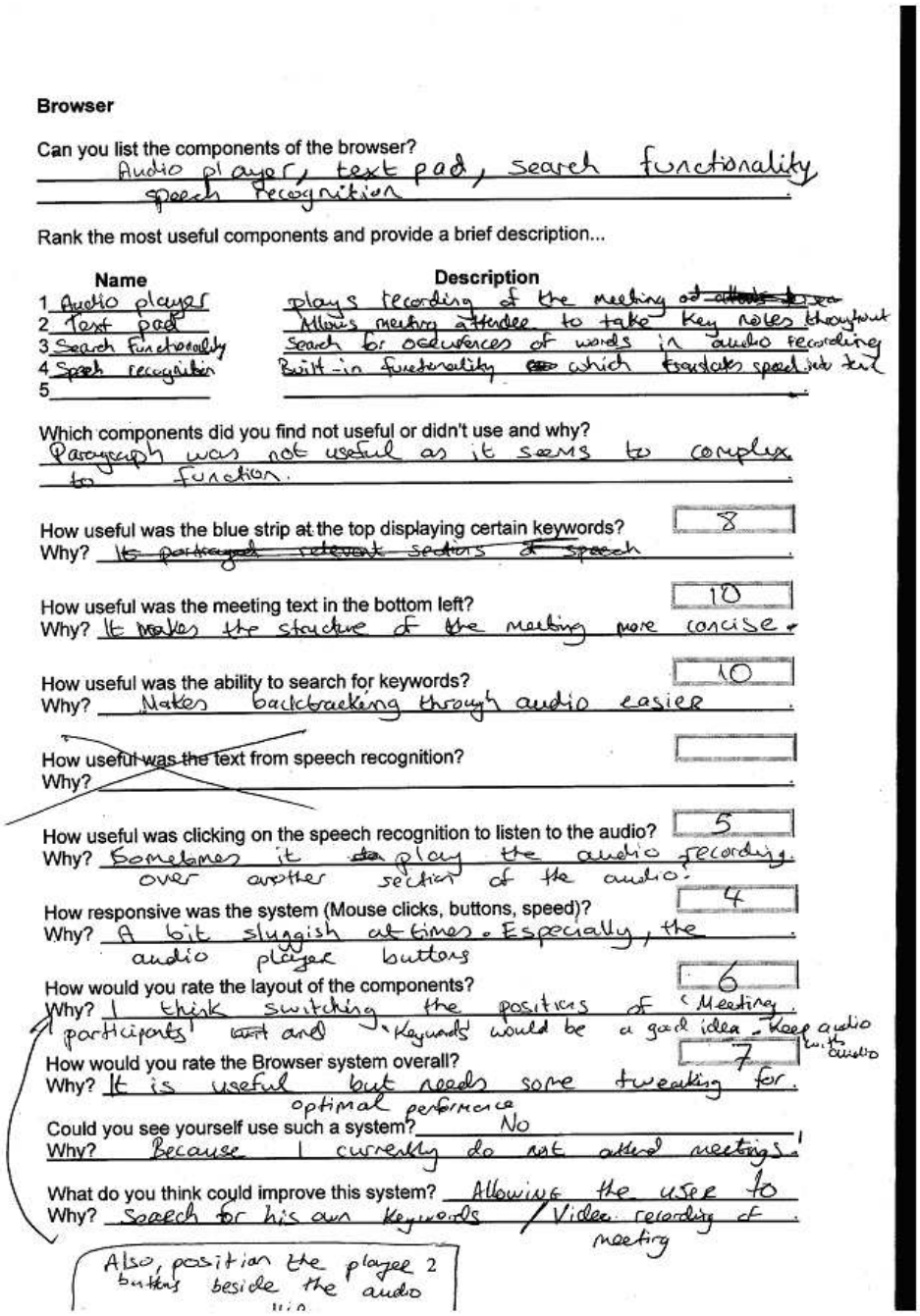
How useful was it to listen to the participants? 10
Why? ~~Very~~ Without audio, it would be useless

How would you rate the quality of the audio? 9
Why? Crisp clear very easy to hear and follow

Rate the audio from totally confusing (0) to really useful? 10
Why?

What do you think could improve the audio?
~~Toggle~~ provide equalizer settings for user to toggle with

Figure E.1: Usability Questionnaire: page 1 of 2



Browser

Can you list the components of the browser?
Audio player, text pad, search functionality speech recognition

Rank the most useful components and provide a brief description...

	Name	Description
1	Audio player	plays recording of the meeting ~~of others to~~
2	Text pad	Allows meeting attendee to take key notes throughout
3	Search Functionality	Search for occurences of words in audio recording
4	~~Speech~~ recognition	Built-in functionality ~~to~~ which translates speech into text
5		

Which components did you find not useful or didn't use and why?
Paragraph was not useful as it seems to complex to function.

How useful was the blue strip at the top displaying certain keywords? 8
Why? ~~Its purpose~~ relevant sections of speech

How useful was the meeting text in the bottom left? 10
Why? It makes the structure of the meeting more concise

How useful was the ability to search for keywords? 10
Why? Makes backtracking through audio easier

~~How useful was the text from speech recognition?~~
~~Why?~~

How useful was clicking on the speech recognition to listen to the audio? 5
Why? Sometimes it ~~to~~ play the audio recording over another section of the audio.

How responsive was the system (Mouse clicks, buttons, speed)? 4
Why? A bit sluggish at times. Especially, the audio player buttons

How would you rate the layout of the components? 6
Why? I think switching the positions of 'Meeting participants' ~~and~~ and 'Keywords' would be a good idea. Keep audio with audio

How would you rate the Browser system overall? 7
Why? It is useful but needs some tweaking for optimal performance

Could you see yourself use such a system? No
Why? Because I currently do not attend meetings.

What do you think could improve this system? Allowing the user to
Why? Search for his own keywords / Video recording of meeting

Also, position the player 2 buttons beside the audio

Figure E.2: Usability Questionnaire: page 2 of 2

Appendix F

Questions of the Information Retrieval Task

Table F.1: Questions of the Information Retrieval Task for Meeting A, part I

1- What is the main reason for choosing the Village over Whelans?
a- larger capacity b- cost of venue is minimal with respect to other costs$\surd$ c- better atmosphere d- better reputation
2- Why are the Thursday and Friday chosen as the nights of the gig?
a- to get a discount from the venue b- to get students on thursday and working people friday $\surd$ c- the band manager's insisted on a minimum of 2 gigs d- because they are the most popular gig nights
3- Why is the first weekend in March chosen for the gig?
a- band are touring in the vicinity$\surd$ b- venue is available c- school holidays d- no other big gigs around that time
4- What is Matt's opinion about the Temple Bar Music Centre?
a- sound quality isn't great$\surd$ b- reputation isn't great c- location isn't great d- too expensive
5- Why was a fourth venue added as a last minute suggestion?
a- management suddenly announced it was available b- the sizes of the first three were rather large $\surd$ c- not as nice as the other three but way cheaper price d- Gary knows the manager
6- What share of the bar money are the organiser entitled to?
a- 0% of bar sales $\surd$ b- 5% of bar sales c- 10% of bar sales d- 20% of bar sales

Table F.2: Questions of the Information Retrieval Task for Meeting A, part II

7- Why were local support bands chosen?
a- to save money $\surd$ b- to attract a local crowd c- to support local talent d- because overseas groups were unavailable
8- Why did the organisers decide to advertise the gig 2 weeks in advance?
a- to save on the cost b- people might forget otherwise $\surd$ c- that's the standard in the industry d- to get people before the end of term
9- What was Matt's first suggestion for advertising on radio?
a- Newstalk 106fm $\surd$ b- Today FM c- 2fm d- Phantom Fm
10- Why does Gary not agree with Matt first radio station suggestion ?
a- had a bad time doing work experience at the station b- incorrect target audience $\surd$ c- too expensive d- they only advertise big events
11- Why is newspaper advertising not used?
a- radio will reach more people b- flyers will be enough c- too expensive $\surd$ d- non specific target audience
12- Why were the Beautiful losers not used as support band?
a- Lead singer is in jail b- They weren't available c- organisers wanted to have two different styles of support band$\surd$ d- they demanded too much money
13- Why is the ticket price raised after the initial suggestion of 15 euros?
a- increased overheads $\surd$ b- to make a higher profit c- high cost of advertising d- people will pay anything these days
14- Why do Matt and Gary want to organize the concert?
a- just to get the band to play in Ireland $\surd$ b- to make some money c- to setup a music promotion company d- to raise money for a student society
15- Why is Polar chosen as the thursday night support band?
a- they've got a gig in Limerick on Friday b- they're recording their album at the week-end c- 2 similar bands will get the whole thing going $\surd$ d- they don't get on with Beautiful Losers

Table F.3: Questions of the Information Retrieval Task for Meeting B, part I

1- Why does Jane think it makes more sense to start the trip in Beijing than Hong Kong?
a- This is part of the travel agency deal b- It's the cheapest c- It would be great do start the trip with doing the great wall $\surd$ d- People in the company insisted on starting the trip from the capital
2- Why does Mathew think they should go to the Forbidden City rather than the Great Wall on the first day in Beijing?
a- The Great Wall is too touristy, Forbidden City is more authentic b- The Forbidden city is really fascinating and should be seen first c- People will be tired from travelling to China $\surd$ d- Trip to Great Wall is organised by Hotel
3- What has Mathew got to say about the Quan Ju De Restaurant?
a- Without doubt the best duck you can eat in Beijing! b- Good but probably overrated and overpriced. c- It's a complete rip off! d- He doesn't know, he's never been there. $\surd$
4- How long does it take to walk from Forbidden City to Quan Ju De Restaurant?
a- 5 minutes b- 10 minutes c- 15 minutes d- 20 minutes $\surd$
5- Why does Mathew suggest renting a Bus from Beijing to go to Great Wall?
a- It 's much more flexible, you can decide what to do $\surd$ b- It is the only way to get to Great Wall c- It would be cheapper than getting taxis for every one d- The train gets crowded and it's a long journey
6- What is the shape of Ping Yao town, according to Mathew ?
a- moon shape b- pear shape c- diamond shape b- tartoise shape $\surd$
7- Which expression does Jane use to describe Kunmig, the capital of Yunan province?
a- modern and cosmopolitan $\surd$ b- dirty and noisy c- run down and unsafe d- exotic and charming

Table F.4: Questions of the Information Retrieval Task for Meeting B, part II

8- Why does Mathew thing they should spend two days in Ping Yao Town if possible?
a- There is so much too see b- To avoid taking two night trains in a row $\surd$ c- It takes two days to walk the surrounding wall d- It's a nice place to chill out
9- What do Mathew and Jane think about Guilin's 1 million population?
a- It's not a lot by Chinese standards $\surd$ b- It's a big city and it might be nicer to stay in a smaller town around there c- They are looking forward all the city's hustling and bustling d- They've heard the mix of various ethnic people is absolutly amazing
10- What is Guilin reknowned for according to Jane?
a- The famous spice market in the city centre b- The great buddhist Pagoda on the outskirt c- The silk museum nort-east of the city d- It's cast peaks doted through the city $\surd$
11- What is Jane's idea about the hiring of a poney or using a cable car to climb Jizu Shan Mountain in XiaGuan?
a- People should have the option because some are scared of poneys $\surd$ b- She thinks taking the cable car will cut the duration of the journey c- She has travelled on poneys before and it's not as fun as you may think d- She's heard that cable cars in China are notoriously unsafe
12- What does Jane suggest doing after Beijing?
a- Get an internal flight to Hong Kong b- Get an internal flight to Ping Yao c- Get an internal flight to Kunming d- Get an internal flight to Shangai $\surd$
13- What does Jane think of staying one day in Shangai?
a- It be good to stay longer as there is so much to see b- One day should be enough $\surd$ c- She'd rather not go to Shangai at all d- It would be a good place to go shopping
14- What does Jane suggest to do to go and visit the Terracota Army from Xian if there is less than 10 people?
a- Hire 2 taxis b- Hire a minibus $\surd$ c- Take the city bus d- Ask Hotel to organise the trip
15- Why does Mathew suggest going through Lunan town?
a- You can visit the bustling market town on the back of a camel b- Because of it's unique traditional and delicious Mutton stew c- The people wear colourful clothes d- It is on the way, just 10 km away $\surd$