

**A Visual Interface for Browsing and Summarizing
Conversations**

by

Shama Rashid

B.Sc Engineering, Bangladesh University of Engineering and Technology, 2004

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF

Master of Science

in

THE FACULTY OF GRADUATE STUDIES
(Computer Science)

The University Of British Columbia
(Vancouver)

April 2012

© Shama Rashid, 2012

Abstract

In our daily lives, we have conversations with others in many different modalities like meetings, emails, chats, blogs etc. At the advent of the Web, the volume and the complexity of the conversational data generated through our day to day communication have increased many folds. A way to deal with this overwhelming amount of interactional information is to use automatic summarization for quick access. Although Machine Learning (ML) approaches can be used to generate automatic summaries, extractive or abstractive, they still have not reached the level of quality of human generated summaries. We introduce here a visual interface that takes advantage of human cognition and perception abilities in conjunction with automatically extracted knowledge concepts for the conversation to analyze it and to automatically generate a summary for it.

Our interface provides the user an overview of the conversation's content and a way to quickly explore it. It aids to identify informative sentences as potential components of the summary based on visual cues. Our objective is to provide the user more control over choosing the topics she wants to appear in the concise resultant overview generated through interactive exploration, thus generating a focused summary. We use an ontology containing nodes for speakers, dialogue acts (DA), and a list of entities referred to in the conversation to provide entry points to the conversation. These concepts in the ontology are derived using classifiers based on generic features making it possible to use the interface to explore any mode of conversational data. In this thesis, we have designed an interface based on the principles of Natural Language Processing (NLP), Human Computer Interaction (HCI), and Information Visualization (INFOVIS) that can be used to browse a human conversation using the mapping of sentences to those ontology concepts and can be

used to generate a brief and focused summary for the conversation. We have evaluated our interface in a formal user study and have found that our interface facilitates widely varying approaches adopted by people trying to analyze a conversation.

Preface

This thesis is based on the work conducted at the University of British Columbia (UBC)’s Laboratory for Computational Intelligence by the NLP group. An overview of the primitive prototype that I started with has been described in Chapter 3. The two stages of redesign of the interface (also discussed in Chapter 3) were done as collaboration among Dr. Giuseppe Carenini, Dr. Gabriel Murray, Dr. Raymond Ng and I. I wrote the codes for the front-end interface and for data parsing while the codes for mapping a conversation to an ontology and for generating an abstractive summary for it had been provided by Dr. Gabriel Murray.

The pilot study and the user study conducted for this thesis, as presented in Chapter 4, were done under the approval of UBC Behavioral Research Ethics Board (BREB) certificate H04-80496. I was the main investigator for these studies and managed all the recruitment. I also administered the pre-questionnaire and the post-questionnaire to the subjects. I was assisted by Anders Linn, an undergraduate student from the Department of Cognitive Science, at the user study who helped in observing the participants during the experiment sessions and occasionally helped in administering the pre-questionnaire.

A version of the first phase redesign was presented at the Visual Interfaces to the Social and Semantic Web (VISSW) workshop of the Intelligent User Interfaces (IUI) conference and published in the proceedings of CEUR-WS.org 2011 workshops with title ‘An Ontology-based Visual Interface for Browsing and Summarizing Conversations’ by Rashid, S. and Carenini, G.

Table of Contents

Abstract	ii
Preface	iv
Table of Contents	v
List of Tables	ix
List of Figures	x
Glossary	xiv
Acknowledgments	xv
1 Introduction	1
2 Related Work	7
2.1 Meeting Browsers and Document Collection Visualization	7
2.2 Word Frequency in a Text Document	11
2.3 Word Repetition Pattern in a Text Document	15
2.4 Syntactic Relationship of Words	20
2.5 Semantic Relationship of Words	23
2.6 Scented Widgets	24
2.7 Evaluation of Summary	27
3 Design and Implementation	29

3.1	Data Abstraction	30
3.2	Stage 1 Design	31
3.3	Stage 2 Design	32
3.3.1	The Ontology View	33
3.3.2	The Transcript View	34
3.3.3	The Summary View	35
3.3.4	Redesign Rationale	35
3.4	Stage 3 Design	37
3.4.1	Display Design	38
3.4.2	Data Encoding	40
3.4.3	Interaction Design	44
3.4.4	Layout Design	50
3.5	Scenario of Use	51
3.6	Abstractive Summary Generation	53
4	Evaluation	56
4.1	Pilot Study	56
4.1.1	Participants	57
4.1.2	Experimental Setup	57
4.1.3	Marking Scheme	59
4.1.4	Results	59
4.2	User Study	61
4.2.1	Participants	61
4.2.2	Experimental Setup	62
4.2.3	A Sample Task	64
4.2.4	Marking Scheme	65
4.2.5	Our Hypothesis	66
5	Results	67
5.1	Summary of Participants' Scores	67
5.2	Revised Summary of Participants' Scores	70
5.2.1	Grouping of Participants Based on Scores Achieved	70
5.3	Pre-questionnaire Results	71

5.4	Post-questionnaire Results	73
5.4.1	Likert Scale Questions on Usability of the Interface	74
5.4.2	Feedback on Summary View and Its Link to the Transcript View	77
5.4.3	Feedback on the Ontology View	78
5.4.4	Feedback on the Entity View	81
5.4.5	Feedback on the Tag Selection Settings	83
5.4.6	Feedback on the Marker Bars	85
5.4.7	Feedback on Accuracy of the Entities Listed in Entity View	87
5.4.8	Feedback on Accuracy of the DA Type Tagging in the On- tology View	88
5.4.9	General Comments	89
5.5	Other Interactional Behaviour	90
6	Conclusion and Future Work	93
6.1	Conclusions	94
6.2	Future Work	94
6.2.1	Color to Distinguish Speakers and Turns	94
6.2.2	Family of Color Shading Scheme for Marker Bars	95
6.2.3	Overview+Detail Approach for Marker Bar	95
6.2.4	Highlight Search Keywords within Transcript	96
6.2.5	Filter Summary Path and Flexible Query Setting	96
6.2.6	Information Scent for the Abstractive Summary	97
6.2.7	Dynamically Adjusting Entity List	97
6.2.8	Better Dynamic Layout	98
6.2.9	Repetition Pattern of Conversations and Clustering of Entities	98
6.2.10	Display Non-linear Conversations	100
6.3	Chapter Summary	108
	Bibliography	109
A	Supporting Materials	115
A.1	Consent Form	115
A.2	Pre-questionnaire	117

A.3	User Study Tasks	119
A.4	Post-questionnaire	122
A.5	Marking Scheme for the User Study Experiment Session and In- structions to the Judges	124
A.6	Pilot Study	126

List of Tables

Table 4.1	Summary statistics for the pilot study	59
Table 4.2	Summary statistics for task wise marking for the user study . .	65
Table 5.1	Summary statistics for scores of the 30 participants	68
Table 5.2	Summary statistics for score of the 28 participants (excluding 2 outliers)	70
Table 5.3	Summary statistics for the post-questionnaire Likert scale ques- tions	75

List of Figures

Figure 1.1	Latest design with 4 integrated views - the Ontology View (left), the Entity View (bottom), the Transcript View (middle), and the Summary View (right)	4
Figure 2.1	Snapshot from the CALO-MA offline meeting browser	9
Figure 2.2	A tagcloud created using the transcript of AMI meeting ES2008a (URL:tagcrowd.com)	11
Figure 2.3	A wordle created using the transcript of AMI meeting ES2008a (URL:wordle.com)	12
Figure 2.4	Placement of a word at the centroid of all appearances along elliptical placement of text in TextArc	13
Figure 2.5	Screen shot of Themail showing a user's email exchange with a friend during 18 months	14
Figure 2.6	Conceptual recurrence plot	16
Figure 2.7	Arc Diagram for different substrings	17
Figure 2.8	FeatureLens	18
Figure 2.9	ThemeRiver of Associated Press data from June-July 1990	19
Figure 2.10	WordTree for Gonzales' testimony in 2007	20
Figure 2.11	Phrase Net for 'X of Y' relation in novel 'Pride and Prejudice'	22
Figure 2.12	DocuBurst of a science textbook rooted at idea.	23
Figure 3.1	Example of mapping a sentence to an ontology	31
Figure 3.2	Stage 1 design	32

Figure 3.3	Stage 2 design with 3 integrated view - the Ontology View (left), the Transcript View (right), the Summary View (bottom)	33
Figure 3.4	Stage 3 design based on InfoVis principles	37
Figure 3.5	Summary View with information scent; the top sub-panel displays an extractive summary while the bottom one shows an abstractive summary for the tagged sentences	41
Figure 3.6	Ontology View concept encoding using size, shape and color channels	42
Figure 3.7	Entity View using size and color for encoding and position as sort order	42
Figure 3.8	Transcript View with separate columns for Speaker and DA type nodes and Speaker incorporated as turn parameter	44
Figure 3.9	Entities sorted by name using control	45
Figure 3.10	Using range slider to shorten list in Entity View	45
Figure 3.11	Marker bar with color encoding core concept and tooltip texts	47
Figure 3.12	Search box to look up keyword	49
Figure 4.1	Test interface with all features enabled	57
Figure 4.2	Baseline interface with DA type nodes on the Ontology View and the Summary View disabled	58
Figure 5.1	Score assigned by judge 1 vs score assigned by judge 2	68
Figure 5.2	User study score frequency distribution	69
Figure 5.3	Average score of the thirty participants shown using a line, points represent individual scores	69
Figure 5.4	Box and Whisker plots for the questions 1 to 9	75
Figure 5.5	Scatter plot for the number of times participants clicked on the extractive summary sentences and their score	77
Figure 5.6	Column chart showing whether the participants from different performance groups found the extractive summary useful . . .	78
Figure 5.7	Scatter plot for the number of times a speaker or a DA type was selected in the Ontology View by a participant and the participant's score	79

Figure 5.8	Column chart showing whether the participants from different performance groups found the Ontology View useful	79
Figure 5.9	Scatter plot for the number of times a speaker is selected in the Ontology View by a participant and the participant's score . .	80
Figure 5.10	Scatter plot for the number of times a distinct speaker was selected in the Ontology View by a participant and the participant's score	81
Figure 5.11	Scatter plot for the number of times an entity was selected by a participant and the participant's score	82
Figure 5.12	Scatter plot for the number of times a distinct entity was selected by a participant and the participant's score	82
Figure 5.13	Column chart showing whether the participants from different performance groups found the Entity View useful	83
Figure 5.14	Scatter plot for the number of times sort order of the Entity View was toggled by a participant and the participant's score .	83
Figure 5.15	Scatter plot for the number of times range slider of the Entity View was used by a participant and the participant's score . .	84
Figure 5.16	Scatter plot for the number of times tag selection settings was toggled and participants' score	84
Figure 5.17	Column chart showing whether the participants from different performance groups found the Tag Selection Settings useful .	85
Figure 5.18	Scatter plot for the number of times markers clicked to navigate within conversation and participants' scores	86
Figure 5.19	Column chart showing whether the participants from different performance groups found the Marker Bars useful	87
Figure 5.20	Column chart showing whether the participants from different performance groups found the listing on Entity View accurate	88
Figure 5.21	Column chart showing whether the participants from different performance groups found the DA type tagging accurate . . .	89
Figure 5.22	Scatter plot for the number of time search button used to navigate and participant's score	90
Figure 5.23	Scatter plot for the number of search terms looked up and participant's score	91

Figure 5.24	Scatter plot for the number of unit vertical scroll used in the Transcript View and participant's score	91
Figure 6.1	Thread View of a conversation thread with 11 posts in 4 levels; the numbering in the diagram traces the route from the root to the node in white labelled 1; the numbering restarts for each level of sibling nodes	101
Figure 6.2	Arc View for the conversation thread presented in Fig. 6.1 . . .	102
Figure 6.3	Dynamic Thread View for the conversation thread presented in Fig. 6.1	103
Figure 6.4	Node Plus structure showing parent-child relation using indentation and sibling relation using the same vertical column position	104
Figure 6.5	Radial Tree with root conversation session at centre and concentric circles indicating level of descendent conversation sessions	105
Figure 6.6	Node Plus structure for displaying transcript of the conversation sessions in the thread	106
Figure 6.7	Narrow Tree for a conversation thread	106
Figure 6.8	Tree Table for a conversation thread where each cell represents a conversation session	107

Glossary

NLP	Natural Language Processing
HCI	Human Computer Interaction
INFOVIS	Information Visualization
RDF	Resource Descriptor Framework
OWL	Ontology Web Language
ML	Machine Learning
NLG	Natural Language Generation
AMI	Augmented Multi-party Interaction
ASR	Automatic Sound Recognition
BREB	Behavioral Research Ethics Board
IUI	Intelligent User Interfaces
VISSW	Visual Interfaces to the Social and Semantic Web
BIN	Business Intelligence Network

Acknowledgments

First and above all, I praise the Almighty for providing me this opportunity and granting me the strength to proceed successfully.

I would like to thank my supervisor, Dr. Giuseppe Carenini for his guidance and support for the last few years. He has not only helped me learn the ropes of academic research but has also always encouraged me to take the challenges of grad life in stride.

Throughout the length of my thesis I have received valuable feedback from Dr. Raymond Ng and he has kindly agreed to be the second reader of my thesis. Dr. Gabriel Murray has been instrumental in every stage of design of the interface. To them I express my gratitude.

I am indebted to my colleague Anders Linn for helping me in preparing the user study materials, in running the studies and in analyzing the results. A big thanks goes to my colleague and friend Shafiq Rayhan Joty for proof-reading the drafts and for directing me to relevant work whenever he happened to come across one.

My thesis has been funded by NSERC through the Strategic Business Intelligence Network (BIN). I would like to express my gratitude to my funding sources.

Last but not least, I wish to thank my parents. An attempt to enumerate how they have supported me in every step of my life would be futile, the list is endless. To them I dedicate this thesis.

Chapter 1

Introduction

In our daily lives, we have conversations with other people in many different modalities. We email for business and personal purposes, attend meetings in person and remotely, chat online, and participate in blog or forum discussions. The Web has significantly increased the volume and the complexity of the conversational data generated through our day to day communication and has provided us with a rich source of readily available public discourse data. This multimodal diverse data has proved to be a challenge from Natural Language Processing (NLP) point of view due to its highly informal nature that render text processing using syntactic, lexicographic and semantic rules often insufficient. Nevertheless, it is clear that automatic summarization can be of benefit in dealing with this overwhelming amount of interactional information by providing quick access. Automatic meeting abstracts would allow us to prepare for an upcoming meeting or review the decisions of a previous group. Email summaries would aid corporate memory and provide efficient indices into large mail folders. Summaries of technical blogs could become an important support platform for developers, administrators, and technology enthusiasts in general. Here we present an interactive interface that takes advantage of human perception in conjunction with automatically extracted conversation concepts to support users in generating a brief, focused overview of a conversation.

Summarization of human conversations have been addressed in the past for different modes of conversation, including meetings [19], emails [11, 39], telephone

conversations [57] and internet relay chats [56]. In all this previous work the dominant approach to summarization has been extractive, which means that the summary is generated by selecting and concatenating the most informative sentences from the source document(s). Extractive summarization has been popular at least in part because it can be framed as a binary classification task that lends itself well to relatively simple Machine Learning (ML) techniques. An alternative is abstractive summarization, where the summary is generated by extracting and aggregating information from the conversation. This approach requires a Natural Language Generation (NLG) component and is preferred by users for coherency but the resultant summary sentences are often too generic and lack the details of the original conversation sentences. Abstractive summarization performs well for relatively small domains where it is possible to enumerate appropriate sentence structures capable of capturing and representing the conversation. This approach is constrained by the richness of its representations and a general-purpose solution requires an open-domain semantic analysis. Extrinsic evaluations have shown that, while extractive summaries may be less coherent than human abstracts, users still find them to be valuable tools for browsing documents [22, 30, 34]. However, these same evaluations also indicate that concise abstracts are generally preferred by users and lead to higher objective task scores. The limitation of a cut-and-paste summary is that the end-users do not know *why* the selected sentences are important; this can often only be discerned by exploring the context in which each sentence originally appeared. One possible improvement is to create *structured summaries* that represent an increased level of abstraction, where selected sentences are grouped according to the entities they mention as well as to phenomena such as *decisions*, *action items* and *subjectivity*, thereby giving the users more information on why the sentences are being selected. For example, the sentence “Let’s go with a simple layout” is about a *simple layout* and represents both a *decision* and the expression of a *positive subjective* statement.

The objective of our interface is to provide the user more control over choosing the topics she wants to appear in the concise resultant overview generated through interactive exploration, aided by a structured visual representation of the conversation; thus generating a focused summary. We use an ontology containing nodes for speakers, dialogue acts, and a list of entities referred to in the conversation to

provide entry points to it. These concepts in the ontology are derived using classifiers based on generic features making it possible to use the interface to explore any conversational data provided (offline) in a text format. However, we have been working mainly on single session synchronous conversations, namely Augmented Multi-party Interaction (AMI) meetings, that have a linear format i.e. the sentences in particular a conversation can be represented in a strictly temporal order along a single thread.

Our first attempt to build an interface to create visual structured summaries of conversations was presented in [10]. This interface relied on mapping the utterances of the conversation into an ontology, similar to the faceted browsers in [16, 55], that then could be used to search the conversation according to the annotation. Our ontology initially contained only the participants of the conversation and properties of the utterance such as whether it was expressing a decision, a subjective statement, etc. and the applied interactive Information Visualization (INFOVIS) techniques were rather primitive. In this thesis, we'll present two iterations of redesign to make the interface more sophisticated and consistent with Human Computer Interaction (HCI) and INFOVIS principles.

In the first redesign, we addressed several limitations of our initial prototype. First, we had extended the ontology to also include entities mentioned in the conversation. Searching the conversation using a particular keyword is suitable only when users already have an idea about the content and want additional information on a particular entity. Representing entities in the ontology enables the users to perform a more refined search and browsing of the conversation and also provides them with a quick overview of the content of the whole conversation. Secondly, we had provided a satisfactory solution to the problem of highlighting the sentences mapped to nodes selected by the users in the ontology. Instead of using color (a non-scalable solution that we initially explored) we used tags associated with the (knowledge concepts within) ontology. The third extension was the addition of a *Summary View* integrating structured visual (extractive) summaries and abstractive focused summaries.

In the second redesign, we improved the interface to enhance its usability based on InfoVis principles; specifically addressing the issues of a better visual representation for the ontology information and an easier exploration of the conversation

transcript. Our latest interface design consists of 4 integrated views - a) the *Ontology View*, b) the *Entity View*, c) the *Transcript View*, and d) the *Summary View* (see Fig. 1.1). To make the interface easily understandable by a wider set of users, we have replaced the NLP jargon with common phrases. So, the utterances are being referred to as sentences, DA types as sentence types, participants as speakers, and entities as topics on the interface.

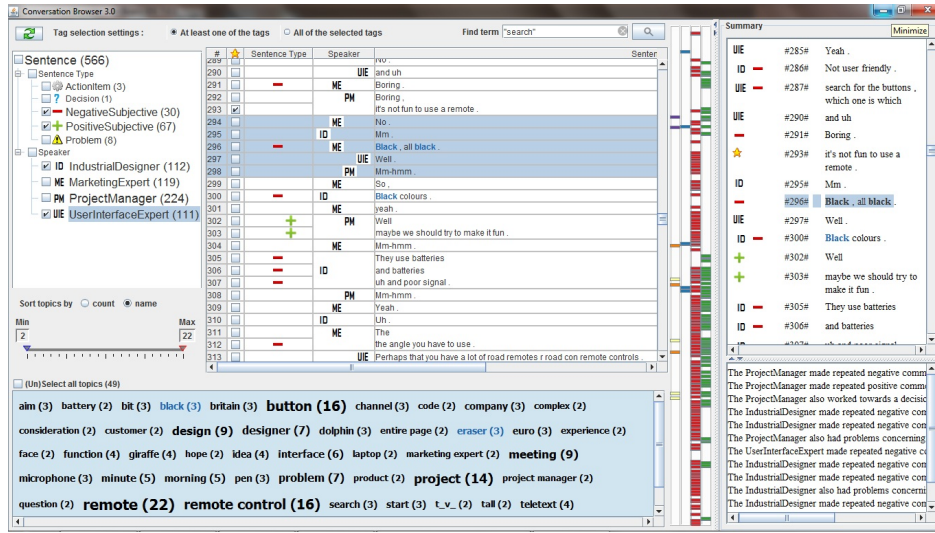


Figure 1.1: Latest design with 4 integrated views - the *Ontology View* (left), the *Entity View* (bottom), the *Transcript View* (middle), and the *Summary View* (right)

The transcript for the meeting conversation that is being explored is shown one sentence per row in the 'Sentence' column of the *Transcript View* (at the centre). Some of the sentences are grouped in larger boxes that correspond to turns taken in the conversation by different speakers. The speaker of each turn is shown in the 'Speaker' column next to the 'Sentence' column in this view. The user can interactively identify important parts of the conversation by using the 'Speaker', 'Sentence Type' or 'Topic' concepts. These concepts are presented in the *Ontology View* and the *Topic View*. The count within parenthesis beside each concept shows how many times that concept appears during the conversation. Whenever a tag is selected from the *Ontology View* or the *Topic View* a marker appears in

the corresponding *Marker Bar* (the 4 vertical bars shown to the right of the Transcript View, blue marker for topic, red for speaker and green for sentence type). The most recently selected tag appears in a darker shade while the tags that were applied before appear in a lighter shaded marker. The user can click on a marker to auto-scroll the transcript to the corresponding sentence. The tooltip text shown when she places the mouse pointer over a marker indicates exactly which concept (*Decision* or *Problem* etc.) the marker was applied for.

The ‘Speaker’ and ‘Sentence Type’ concepts are shown in the *Ontology View* (top left). When the user selects a sentence type e.g. ‘Decision’ the corresponding icon (the blue question mark) appears in the ‘Sentence Type’ column of the *Transcript View* for each line that contains a sentence that expresses a decision. The topics or phrases that are referred to over the length of the conversation are shown in the *Entity View* (bottom). If the user selects one of the topics from this view it gets highlighted in the ‘Sentence’ column of the Transcript View within the sentence itself in a bold blue color. The ‘Sort by Count or Name’ control for the *Topic View* can be used to find out what was the most discussed about topic or whether a particular topic appears on the list. The user can also employ the range slider provided to make the list of topics displayed shorter and easier to concentrate on. Alternatively, the *Search Box* above the *Transcript View* can be used to find any keyword specified in the input field.

A summary generated for the tagged sentences appear in the *Summary View* (right); the top summary panel shows all the tagged sentences and their corresponding tags, the bottom summary panel shows a more natural sounding summary generated by combining related tagged sentences. The user can click on any sentence in the Summary View to check the sentence(s) in the original transcript corresponding to it. The ‘Tag Selection Settings’ menu (top) can be employed by the user to specify whether she wants to investigate the sentences that contain at least one of the selected tags or the sentences that contain all of them (normally results into a smaller set of sentences).

As a third contribution of this thesis we have designed and run a user study in which we asked the participants to browse a conversation aided by our interface. The users were requested to complete a set of tasks that posed queries about different topics discussed during the conversation. Throughout the entire session we

automatically logged all interactional behaviour like the number of times our participants used different components of the interface and the answers submitted by the participants for the assigned tasks were judged manually for relevance.

To summarize, in this thesis we present a visual interactive interface to create focused summaries of human conversations. Our interface allows the user to explore conversations and to identify informative sentences based on their association with nodes of interest on the visual representation of an ontology. The sentences thus selected as potentially important components of the summary can then be further inspected by the user and through iterative refinement of ontology concepts derive a brief and focused overview of the conversation. In Chapter 2, we discuss approaches taken by different meeting browsers and document collection visualization tools to facilitate browsing in an organized way. We also discuss different document visualization aspects and conversation thread visualization options for our future extension to non-linear conversations. In Chapter 3 we shall discuss in details the two stages of redesign we implemented for our interface and the rationale behind them. Chapter 4 discusses the user study we ran to evaluate our interface and we present our findings from that study in the following Chapter 5. Finally, we discuss future extensions to the interface based on our findings during the evaluation.

Chapter 2

Related Work

In this chapter, we take a look at a few other interfaces used to browse meeting conversations and to visualize document collections. We also discuss literature related to different visualization and interaction components of our interface like representation of the entities and navigation means for the transcript.

2.1 Meeting Browsers and Document Collection Visualization

The idea of using an ontology to explore data in an orderly manner is not novel. For instance, the Flamenco [55] and the Mambo [16] systems make use of hierarchical faceted metadata for browsing through image or music collections. In our approach we adopt similar techniques to support the exploration of conversations. More specifically, in Flamenco [55], while navigating an image collection along conceptual dimensions or facets (e.g. date, theme, artist, media, size, color, material etc.), every facet hyperlink that can be selected to derive a new result set is displayed with a count as an indicator of the number of results to expect i.e. the count works as a query preview. Similarly, we have included a count beside each node of the ontology to indicate the number of sentences in the conversation that have been mapped to it.


An extractive approach for generating a decision-focused summary suitable for debriefing tasks has been proposed in [24]. This type of summary includes only 1-

2% of a meeting recording related to decision making. In addition to the transcripts, the interface takes advantage of the audio-video recordings to better understand decision points. While the interface in [24] makes use of only dialog acts for focused summary generation, ours additionally uses speaker and entity information. Furthermore, we are not limited to extractive techniques as our interface decouples the task of content selection and summary generation and thus provides the user option to choose an abstractive technique. The interface proposed in [24] also considers features that are specific to conversations about designing a new product (see AMI corpus [13]), in which you typically do not have only a single meeting but a series of meetings, the kickoff, the conceptual design, the detailed design, and the evaluation meetings. While we also aim to consider series of related conversations we intend to do it in a general way, i.e., not being limited to conversations about designing a product.

The Ferret Meeting Browser [52] provides the ability to quickly find and play back a combination of available audio, video, transcript and projected display segments from a meeting side by side for comparison and inspection synchronously and allows navigation by clicking on a vertical scrollable timeline of the transcript. Users can zoom into particular places of interest by means of a button and by zooming out they get an overview of the meeting in terms of who talked the most, what meeting actions etc. The Marker Bars on our interface is similar in concept to the Ferret timeline showing concentration of a concept along the length of the conversation.

The Meeting Miner [9] aids browsing multimodal meeting through recordings of online text and speech collaborative meetings using timeline navigators of content of edits as the main control for browsing. In addition, it can retrieve a set of speech turns spread throughout the conversation focused on particular keywords that can be typed in or selected from a list of automatically generated keywords and topics. The browser also facilitates interactive navigation among these segments identified as being relevant to the search. The users can also navigate to the audio segments that have been identified as relevant using the audio timeline for random access of the file. The Meeting Miner [9] automatically identifies a set of potential keywords and the users can decide to view these in alphabetical order, ranked by term frequency or simply by time of appearance in the conversation. A

similar concept has been discussed in the future work of FacetMap [42] where the authors mention implementing the ability to dynamically order the facets, such as by count, alphabetically by label, by usage, or by some specific facet ordering. The entities on of our interface are equivalent to Meeting Miner’s keyword panel entries and we are currently listing the entities ordered based on count by default and are providing users the option to sort by alphabetical order.



Summary	Transcript	Action Items	Topics	QA Pairs	Ink	Meeting Notes	Mark Meeting	Ink 2.0	Timeline
00:10	Donald Kintzing	let's go .							
00:38		just mean you know i guess .							
00:43	Clint Frederickson	trying to these other guys to to join up there were just waiting for collab pop in .							
01:08		yeah .							
01:08		they're really really quick and let me give you an update on the e. p. server situation .							
01:14	Lynn Voss	okay .							
01:14	Clint Frederickson	um so uh kind of working on this on two fronts uh we're trying to figure out why we can't seem to make it connection to it uh kind of on the friends in front uh see if some changes in made or something happened .							
01:31		and then on the mercury fronts uh what is going to protect ourselves better from from this kind of situation by setting some better timeouts and probably executing that call in a nothing thread in an asynchronous matter .							
01:49	Lynn Voss	okay .							
01:50	Clint Frederickson	uh well i really i haven't talked to my friends in this morning .							
01:51	Lynn Voss	so is mike working on it now or is this just on a to do list ?							

Figure 2.1: Snapshot from the CALO-MA offline meeting browser

The CALO meeting assistant [44] is used for capturing audio signals and optional handwriting recorded by digital pens for distributed meetings (see Fig. 2.1). During the meeting the system automatically transcribes the speech to text and the participants are fed back a real-time transcript to which annotations can be attached. At the end of the meeting the system performs further semantic analysis on the transcript like dialog act segmentation and tagging, topic identification and segmentation, question-answer pair identification, addressee detection, action item recognition, decision extraction and summarization (see [27] for an introduction to most of these NLP tasks). The result of this analysis is made available

via a web-based interface. The off-line meeting browser interface displays the meeting transcript segmented according to dialog acts. Each dialog act is shown along side its start time, speaker, and a link for streaming audio feedback for the transcript segment (in case the users want to overcome any speech transcription errors). The CALO browser also provides the users views of the extractive summary of the meeting and above mentioned annotations in separate tabs. A lot of the annotations provided by the CALO system overlap with our segmentation of the transcript and knowledge concepts represented in the ontology tree but the CALO browser provides more flexibility by providing the users means to attach their own annotations, which is an interesting direction we could explore in our future prototypes. Our interface differs from CALO by providing a way to focus on the users' particular information need by referring to the ontology and by providing an option to generate either abstractive or extractive summaries.

In iBlogVis [25], the authors use social interaction cues like comment length, number of comments, regular commenters etc. and content cues like topics of a blog, blogger's posting habits etc. to provide the users with an overview of a blog archive and to support them in deciding which entry to read. The font size of a tag for blog topic representation indicates its popularity, a concept that we have employed for our textual collage representation of conversation content in the Entity View (Section 2.2). iBlogVis uses the idea of read wear [23], a means of graphically portraying the document's readership history, to help users keep track of entries that have been read, have not been read, or the one that is currently being read using different colors. This is an interesting idea we have not implemented in our interface but in future work we could similarly provide users an option to log the current ontology settings so that they can keep track of the combinations tried before.

MostVis [41] uses a multiple co-ordinated view for browsing a catalog for multimedia components in a car. Besides the textual label of each node in the catalog node-link tree representation there is an additional icon representing element type (car series, function block, functions, parameters etc.). This inspired our own use a short string representation or icon beside the ontology tree nodes. MostVis also has a history window with undo and redo buttons where an entry is logged every time an expansion or minimization of the node-link tree occurs. We are exploring

how a similar mechanism could be added to our interface.

2.2 Word Frequency in a Text Document

The distribution of words or phrases in a text document has been explored using different techniques; the central theme of all of them being the use of size to encode the frequency of occurrence of a word.

The simplest visual representation of tag or keyword usage statistics in a text document is a tag cloud [2]. A tag cloud uses font size to encode importance of each keyword or tag and lists the tags contiguously in a line, typically in alphabetical order. For such an ordering, font sizes vary widely within a line and place large and small text interspersed, wasting a significant amount of screen space as white space in between lines. Due to this wasteful use of white space and monotonic font style, tag clouds are not considered to be aesthetically pleasing. Tag clouds also disregard the relationship among tags while placing them and do not address distribution of the keywords along the length of the conversation. Despite these limitations, we decided to use a tag cloud like representation for our Entity View because of its simplicity and its suitability as a tool for analysis of a document and for comparison across documents.

Figure 2.2: A tagcloud created using the transcript of AMI meeting ES2008a (URL:tagcrowd.com)

Wordles [47] use tag-cloud-like displays that give careful attention to typography, color and composition. The font size of a word is linearly related to frequency of that word in the text. Wordle makes compact use of space and are more engaging than tag clouds. However, wordles cannot be used to compare contents of documents due to irreproducibility caused by the randomness in the placement algorithm and due to non-uniform scaling of font across documents making wordles ineffective as analytical tools. In an effort to get a more compact visualization,



Wordles places the text in vertical, horizontal or diagonal layout, which affects legibility of the text.

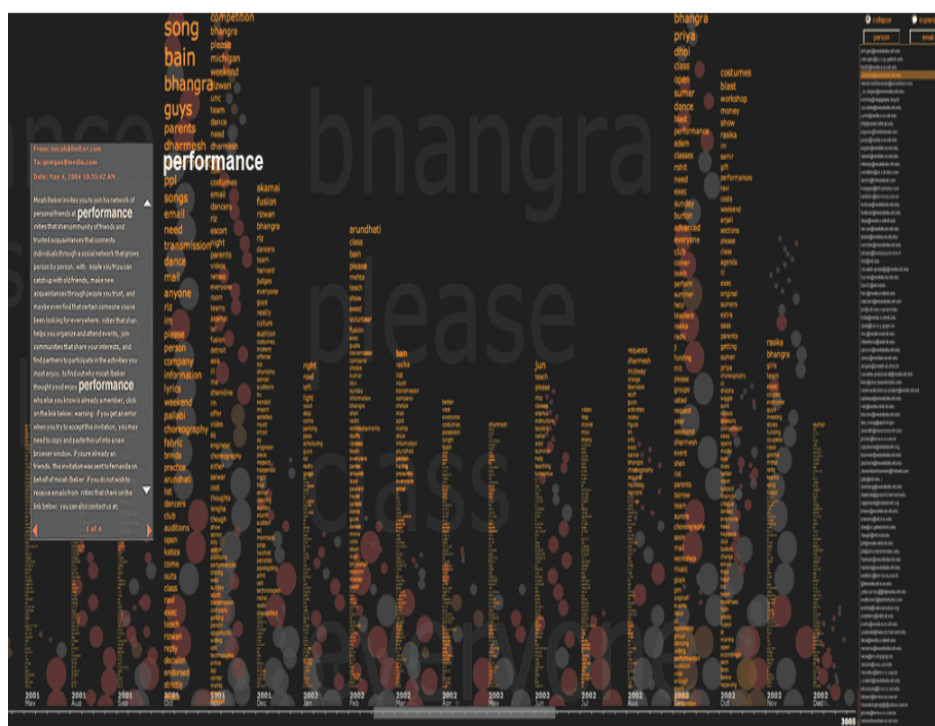


Figure 2.5: Screen shot of Themail showing a user’s email exchange with a friend during 18 months

word is based on its frequency and how distinct the word is to specific relationship. The month and year are shown at the bottom of the columns. The monthly words are interactive; clicking on a word opens the relevant message in a details view. Each email is shown as a circle with size proportional to the length of the message, and with color indicating whether the message was incoming or outgoing. Themail’s use of related family of colors to encode message direction and frequency of words can be misleading to the users. Themail provides an option for visualizing the email archive in a collapsed view where the months without any correspondence are not displayed ensuring better use of available screen space. This dynamic space allocation concept is something that was addressed in our interface but there is a scope for further improvement. A problem with Themail is that quoted text within an email thread impacts the frequency of keywords. Another limitation is that it is capable of handling only individual words but not phrases, a

limitation that has been addressed in our interface by including noun phrases in the Entity View. iBlogVis presented in [25] also uses a tag cloud representation of the heavily discussed topics.

2.3 Word Repetition Pattern in a Text Document

The distribution of frequently appearing words or phrases along the length of the text document conveys more information about the structure of discussion and indicates parts of the text that may prove more informative than others. A dotplot matrix [14] is the simplest approach to represent repetition pattern in a text string. It uses a matrix where a dot is placed at row i and column j if the i th character in the string and the j th character in the string is a match. The matching can be done at different granularities like character, word, or any substring. Although dotplots are able to handle arbitrarily long strings, the quadratic scaling makes it highly inefficient and hard to interpret. Another limitation of dotplot matrix is that it redundantly repeats the same match pattern mirrored along the diagonal.

Conceptual recurrence plots [7] are used to visualize global and local patterns within time series data where data similarity at different time points in the same time series of data is computed on a conceptual level and shown extending the idea of dotplots. Like dotplot matrix, conceptual recurrence plot places the data along the X and Y axes of the matrix and the temporal flow is encoded by the placement of a data point along these axes, from left to right or from top to bottom (see Fig. 2.6). Since the data similarity is not a boolean match anymore rather a quantitative score, color gradient or shading is used instead of solid colors as encoding channel. Only the lower triangle of the matrix is plotted to avoid mirroring of visual match features and the most highly recurring concepts are shown in a separate panel similar to our Entity View. Different people have different vocabulary and may use different terms to refer to the same semantic construct which is taken into account by conceptual recurrence plots. Definition of the basic unit of text representation or utterance is domain specific and block sizes in the recurrence plot is optionally made proportional to actual time spent on talking. The plot can be enhanced by using different colors for specific speakers, groups, types or other categorization within conversation. This type of recurrence plots is useful for analyzing pattern

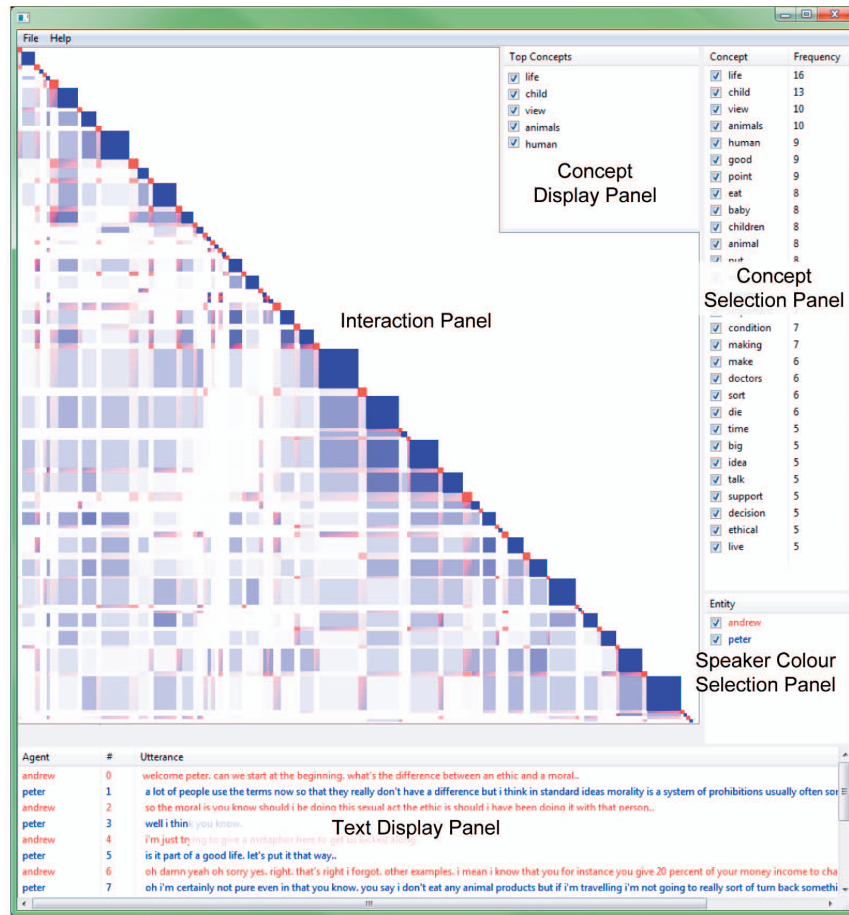


Figure 2.6: Conceptual recurrence plot

of communication like degree of interactivity of a pair of participants or location of key concepts. We can use them in the future to analyze recurrent portions of meeting conversations in a series or among posts in an email or blog thread.

TimeSketch [5] uses half disks to link different sections of a piece of music that show repetitions. It also colors related passages with the same color to indicate musical form. TimeSketch does not scale well for sequences that have many different related passages and the relevance of the passages have to be annotated manually which is cost intensive.

Arc Diagrams [49] are used to represent complex patterns of repetition in string

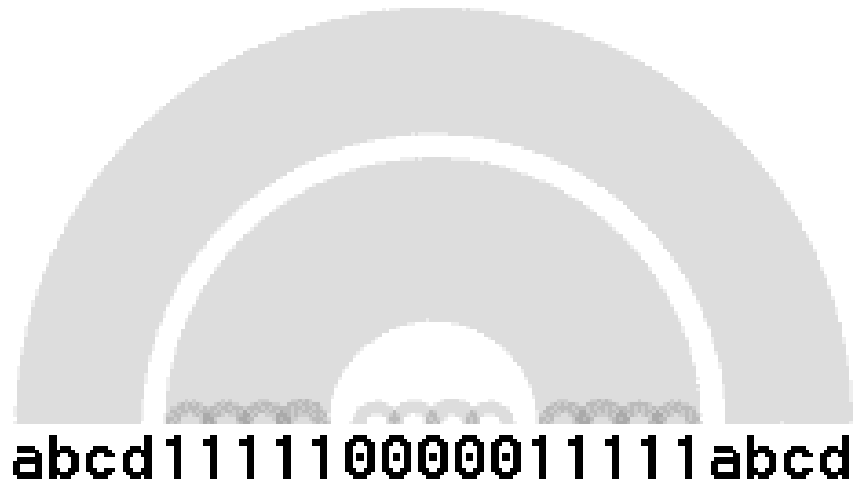


Figure 2.7: Arc Diagram for different substrings

data with many different repeated sub-sequences and multiple scales of repetition. For each consecutive matching pair on the string, the corresponding intervals are connected with a translucent thick semi-circular arc. The height of the arc is thus proportional to the distance between the two sub-strings and the line width of the arc corresponds to the length of the substring under consideration. The translucency allows revealing highly repetitive structures since overlap of patterns now appear with more opacity. The authors used arc diagrams to explore the repetition patterns in music and program codes like HTML or a Java class. For a human conversation, there is similar collocation and repetition pattern for phrases and words pertaining to a topic. On the marker bars of our interface (see section 3.4.3) though we were unable to use different colored markers for each entity, speaker or DA type due to scalability issues, the most recent selection for each concept type is shown using a darker shade (blue for entity, red for speaker, green for DA type). The distribution of the markers along the marker bar gives a high level idea on how a particular concept is dispersed over the length of the conversation. The juxtaposed position of the marker bars for the speaker, DA type and topic concepts make a visual interpretation of intersection of concepts possible. Arc diagrams are suitable for visualizing only a subset of all possible matching substrings and are more efficient than dotplots since only consecutive pairs are connected using arcs

and not every possible pair.

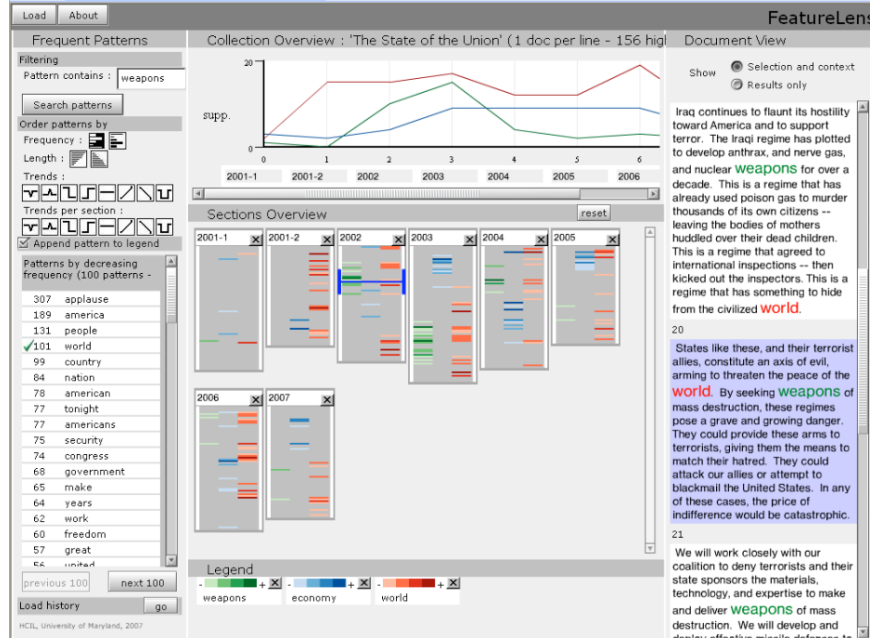


Figure 2.8: FeatureLens

FeatureLens [17] provides a list of highly repeated phrases in the document and uses gradient coloring to indicate frequency of occurrence of each pattern in a paragraph of the document. It provides a visualization of a collection of texts at different levels of granularity, both at the document level, using rectangular panels to represent each document; and at the paragraph level, using colored lines within the panels to represent paragraphs related to each selected pattern. The more frequently the pattern appears in the paragraph, the more saturated color is assigned to the line. The distribution of these lines for a particular pattern is similar to our distribution of markers along a marker bar (see Fig. 3.11). When multiple patterns are selected for inspection, they are assigned different colors and the lines are placed in juxtaposed columns. Due to distance imposed by intervening columns, it is difficult to pinpoint a part of the document where a combination of patterns occurs simultaneously. This task is facilitated by the Collection Overview panel where a line graph is shown for the selected patterns. In our interface, all patterns

of a particular concept type are shown along the same vertical marker bar making it easier to pinpoint the co-occurrence of patterns using the density of markers. However, the user has to rely on the tooltip texts to identify the exact concepts if a large number of concepts of a particular type are selected simultaneously.

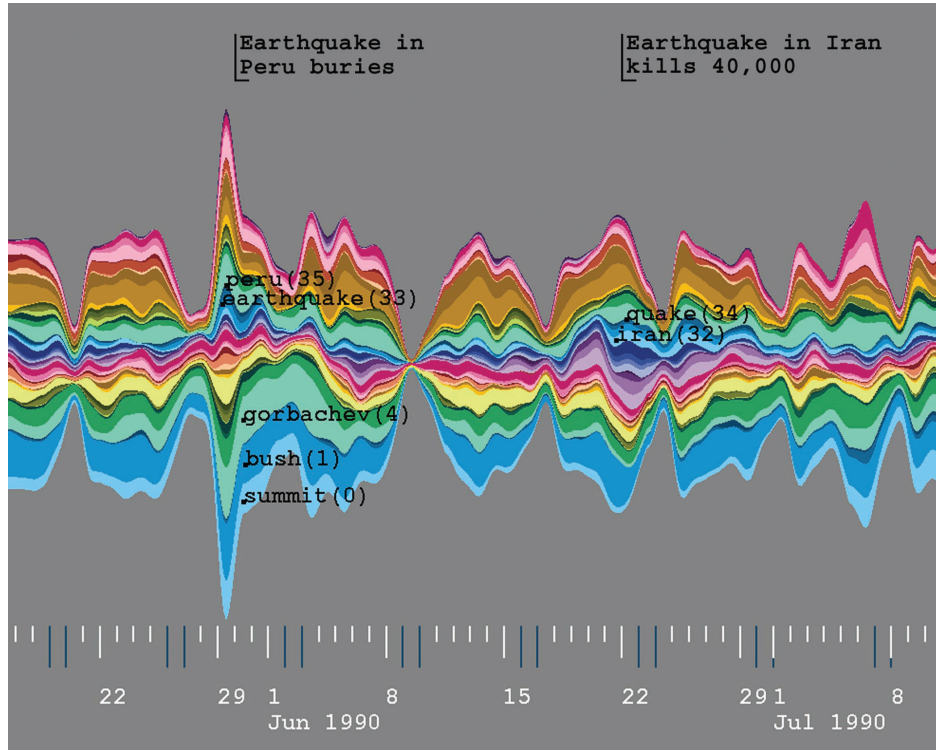


Figure 2.9: ThemeRiver of Associated Press data from June-July 1990

ThemeRiver [21] is a visualization of temporal thematic changes for large collection of documents. Sudden change in the river indicates external event or causal relationship. The timeline is indicated by the directed flow in the visualization, the selected thematic content by composition of the colored current, and thematic strength by changing width. Horizontal distance along the flow direction corresponds to a time interval and the river may dry up at some point in time since only a selection of themes is shown (see Fig. 2.9). A theme current remains the same color for the entire length of the river which imposes a limitation on the number of themes that can be shown without losing discernibility. Also, color perception

depends on local contrast. The stacking effect of themes may hinder perception of their strength. ThemeRiver also employs scented widgets like histograms for the percentage or ratio of document included in the display, and the number of documents mentioning theme or number of occurrences of related words. Short descriptions of external events are shown along the timeline for ease of interpretation. The authors mention using a color family for a group of themes in future extensions; this is similar to our idea of using a two-shade scheme for different concept types. Since dates are not continuous data, interpolation to get a smooth river-like curve may mislead the user.

2.4 Syntactic Relationship of Words



Figure 2.10: WordTree for Gonzales' testimony in 2007

The collocation of words based on lexical or syntactic relations is another aspect of text content analysis. WordTree [50] is a visualization used for quickly querying text document to find repetitive contexts (see Fig. 2.10). It is a graphical display of keywords in context but is unsuitable for providing a full document overview or overall word frequency. It largely preserves the linear arrangement of text and uses font size to represent the number of times a word or phrase ap-

appears. The size of the word is proportional to the square root of the frequency of occurrence in contrast to the most text visualization techniques that use a size linearly proportionate to the frequency of occurrence. Using the square root rather than a linear scale makes the area of the word is very roughly proportional to the frequency disregarding the variation created by word length. WordTree shows the words that follow a particular search term in a tree like structure alleviating the difficulty of seeing pattern in an array of text, the branching of the tree continue until they define a unique phrase. The branches may be ordered in three alternate ways - by frequency of the phrase, alphabetically or by order of occurrence in the text. Our interface also provides the flexibility of listing the entities in the Entity View by frequency or by lexical order. The word tree does not provide any sort of overview of the text nor does it present an initial search term for viewers to start from i.e. there is no natural entry point to the text. Unless the user has a general idea about the text content and has some specific keywords of interest, this would make browsing through the text hard. The authors mention including some tag-cloud like structure as a possible solution to provide an initial entry point similar to our Entity View as natural entry point to the conversation. The word tree tracks the sequence of actions like a web browser does. This allow the user to review her previous steps in visualization by clicking on a browser like back and forward button. A similar back and forth navigation control for the generic keyword search was requested by one participant during our user study. The Word Tree provides users a highlighter mode where a user can add a comment by clicking on a word. We can extend the override control in our interface and provide the user a footnote insertion capability which would be useful especially for multiple session analysis of the same file. A common request from the WordTree users was for the ability to click on an item in the visualization and see the places in the raw transcript where the item appears by drawing lines from the WordTree to a vertical line representing the extent of the text. This is the same as our approach of highlighting the entities in the transcript when that entity is selected in the Entity View. In addition, we have chosen to use a two-shade color scheme to distinguish the entity chosen by the current action (dark blue markers) from the other entities selected (markers shown in light blue).

The Phrase Net [18] interface allows users to input different regular expres-



Figure 2.11: Phrase Net for ‘X of Y’ relation in novel ‘Pride and Prejudice’

sions to analyze a single text document from different perspectives. For instance in figure 2.11 the text from Jane Austen’s novel ‘Pride and Prejudice’ has been analyzed based on ‘X of Y’ relation which reveals a cluster of the central characters (at the top left corner of the resultant graph) with the female protagonist ‘Elizabeth’ placed at the centre. In our work, the concepts like speakers, entities, DA types act as natural entry points, but we are not handling sentence level analysis of word relations. The phrase net merges word nodes that are topologically equivalent to a single node regarding their relationship with other words in the text. For such clusters containing multiple terms, the authors render all terms in a vertical

list, scaling each term individually based on their frequency of occurrence. For example, in figure 2.11 both the words ‘kindness’ and ‘common’ are related to the word ‘humour’ and have no other neighbours, making them topologically equivalent, and so were displayed in a vertical list. The authors also encoded the ratio of out-degree to in-degree for a node and rendered a node with a higher value in a darker color. This helps users to spot which terms occur in the first part of a pattern or last. This interface allow users to set a maximum number N of nodes to show the most relevant ones similar to our range slider control.

2.5 Semantic Relationship of Words

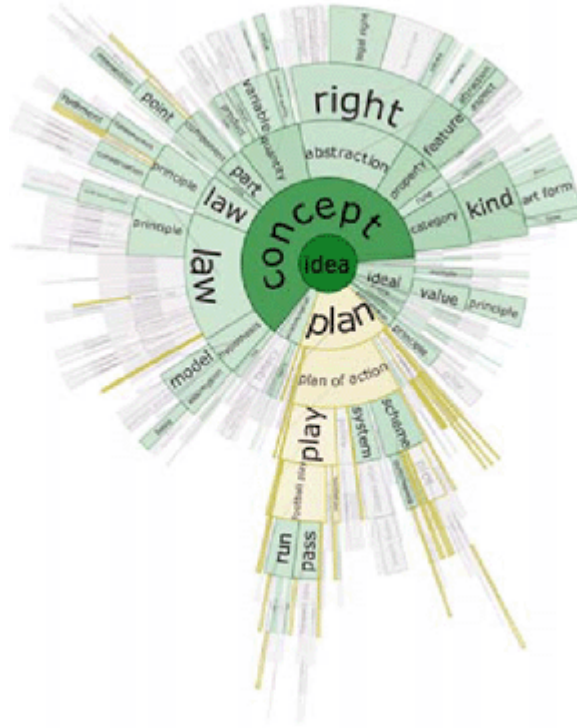


Figure 2.12: DocuBurst of a science textbook rooted at idea.

Most document visualizations do not take into account word meanings or semantics. DocuBurst [15] combines word frequency with human created lexical structure to create a visualization based on semantic content (see Fig. 2.12). The

resultant glyph is a radial, space-filling layout of hyponymy (X is-a Y relationship, for example an apple is a fruit). DocuBurst provides cross-document summary of text for comparison at a glance. The nodes are sized based on leaf count or are made proportional to the sum of word counts for synsets in the subtree rooted at that node. For instance in figure 2.12, the node size based on leaf count gives a consistent representation across documents, synset shape and coloring differs across documents occurrence count of words in subtree. Hue of color is used to indicate synset or set of words with related meanings; highly opaque nodes have high frequency of occurrence while transparent nodes appear rarely. Nodes matching a query are highlighted in gold and a separate color is used to trace a concept upto its root concept. DocuBurst facilitates interactive analysis using geometric and semantic zooming, a detail in context view, drill down within text, linked access to source text.

2.6 Scented Widgets

Scented widgets [53] are graphical user controls that rely on embedded visualizations to facilitate navigation using cues based on a user's perception of value, cost or access path of information sources. Semantic navigation makes cues based on various forms of metadata or the content itself whereas social navigation provides cues based on usage data. Since widget sizes, shapes and layouts are typically fixed, only a few of the visual channels (hue, saturation, lightness, texture etc.) can be used without interfering with the usability of the widget. The count beside the ontology concept labels, the marker bar, the sentence tags in the summary view, color saturation to show recency of concept selection are all examples of scented widgets that provide cues for more informative data. Additionally, we could have used a histogram slider using histograms to show the distribution of entities with each frequency by embedding our entity range slider with similar information.

The participatory information presented in the Ferret meeting browser [52] is similar to the distribution of markers along the Speaker marker bar in our interface. [51] uses a grid structure to show how the metadata of the text corresponds to a row in the text. Each paragraph in the document overview is lined up with a column in the grid and metadata concepts correspond to the rows. To show that a tag

has been applied to a particular paragraph the cell where the appropriate column and row meet is shaded. The color of the cell is determined by the value of the applicability attribute. When a mouse hovers over a grid cell all relevant tags on the metadata tree are highlighted. The metadata is another example of facets of the conversation text and the grid is an alternative to our marker bar in showing distribution of concepts.

Window frames can be used to display contextual information about a document without interfering with the presentation of the document content [8]. The interface designer can use basic design variables like color, thickness, texture and shape of the frame without increasing visual complexity. Moreover, a more complex visual cue may be provided by mapping part of document to part of the frame and these cues may be modified dynamically when plotting results of processes and functions in the frame. A common approach is to map the vertical side of the frame to the length of the document. In the application ReadabilityViewer [8], authors use the right side of the frame to show color gradient to indicate a readability index for the passages of the text displayed in the window. The authors also mention using the left side of the frame to display what is presently being displayed within the window. This is similar to my idea of magnifying the markers for a selected part of the document length upon selection by the user. ScrollSearcher [8] maps the result of searching a text string within the document to the frame and uses different colors and tooltip for different searches. As all the searches are shown in parallel frames next to each other they can be compared, and compound search is made possible. In our interface, the large number of concepts used as entry points to the conversation made it impossible to get a scalable solution where different colors would be assigned to different search terms or concepts. However our solution of the placement of the four marker bars side by side still makes it possible to do compound searches similar to the concept presented here.

The concepts that appear along the length of a conversation can be grouped according to collocation, word meaning or high level topics of the conversation. Although the clustering approach for concept identification may or may not produce the term types and granularities useful to the user, it may be extremely beneficial in reducing load on human cognition. So, as future work we are planning to investigate the presentation of the entities in our interface in logical groups. We discuss

some possible approaches to clustering entities here.

[29] presents a semi-supervised algorithm that uses a root concept, a basic level concept and recursive surface patterns to learn hyponym-hypernym pairs automatically from the web, position the concept based on web query and a graph-based algorithm that derives from scratch the integrated taxonomy structure of all the terms. Pattern based approaches are highly accurate but require a set of seeds and well-defined surface patterns. It also requires a large corpus like the web. Different tasks and criteria produce different taxonomies, even when using the same base level concepts. Attempts at producing a single multi-perspective taxonomy fail due to complexity of interaction among the perspectives. The major problem in performing taxonomy construction from scratch is that overall concept positioning is not trivial. This algorithm uses doubly-anchored lexico-syntactic patterns and bootstrapping to harvest terms and thus can adapt easily to different domains requiring minimum supervision. Once the terms are harvested a hypernym relationship direction between pairs of related terms is established by using a set of surface patterns and from the web hit counts the direction of the relationship is established.

[54] incorporates techniques from text mining, information retrieval, natural language processing and machine learning to generate a concept ontology. Nominal N-gram mining is used to identify the concepts. WordNet and surface text pattern matching are used to identify the relationships among the concepts. A supervised clustering algorithm is then used to further cluster the concepts based on pseudo-relevance feedback. Nominal N-gram mining consists of sentence segmentation, POS tagging, identifying sequence of nouns, mainly bigrams and trigrams and proper nouns where each word starts with a capitalized letter. The noun phrases with highest frequencies in the document are chosen as candidate noun phrases and their validity verified using a web query and threshold of hit count. The bigram concept candidates are organized into groups base on the first sense of the head noun in the WordNet. Trigrams are then compared with bigrams already in the hierarchy. If a bigram concept matches with the suffix of a trigram or named entity, the trigram or named entity is added as a child to that bigram. Webquery is then used to identify hypernym-hyponym relations among sibling concepts. The pairwise similarity score can be a linear function of some underlying features such as

similarity of Web definitions of the two instances, similarity of sub-concepts, similarity of verb usage of the two instances etc. K-medoid clustering with sampling is used to get the final hierarchy tree. A web-based approach is used for labelling the intermediate node concepts found in clustering.

2.7 Evaluation of Summary

Evaluation techniques for summaries can generally be classified as intrinsic or extrinsic [26]. Intrinsic metrics evaluate the actual information content of a summary. Usually the comparison is made with a gold-standard human summary or with the source text. Extrinsic evaluations are done to assess the usefulness of the summary in performing a real-world task; that is why they are sometimes called task-based evaluations. Most summarization work depend more on intrinsic measures than extrinsic measures because such evaluations are easily replicable. However, evaluating summaries intrinsically is labour intensive and subjective since there isn't a single best summary for a given text necessitating a number of human annotations for better judgement. Again, all summarization work is ultimately done for the purpose of facilitating some task and thus should be evaluated in the context of that task. Refer to [12] for a comprehensive list of intrinsic and extrinsic techniques used to evaluate summaries. In a relevance assessment task to extrinsically evaluate a summary, a person is provided a description of a topic or event and then must decide whether a given summary or source text is relevant. In a reading comprehension task, on the other hand, the user is given either a full source or a summary text and is then given a multiple-choice test relating to information from the full source. One can then compare how well users perform in term of the quality of their answers and the amount of time to produce them, when given only the summary compared with the full source document. This evaluation framework relies on the assumption that truly informative summaries should be able to act as substitutes for the full source document.

In [35] a user study was conducted to evaluate abstracts of AMI meeting conversations; three types of summaries were included for comparison - gold-standard human extracts, gold-standard human abstracts, and automatic abstracts. During the study, participants were asked to browse the meetings in order to understand

the gist of them within a time constraint. The participants were asked to consider the scenario in which they were a company employee who wanted to quickly review a previous meeting by using a browsing interface designed for this task. Upon finishing their review of each meeting, participants were asked to rate their level of agreement or disagreement on several Likert-scale statements relating to the difficulty of the task and the usefulness of the summary. We have based the post-questionnaire of our user study on this work.

Chapter 3

Design and Implementation

Our prototype has been developed in 3 major phases, using Java Swing and AWT components and Jena, an open source Java framework that provides a programmatic environment for building semantic web applications. We have also used Python scripts to do the data processing at the backend of the interface. In the first stage (see section 3.2), we had developed a rudimentary interface as proof of concept that conversation knowledge concepts can be successfully used as entry points to the conversation. In the second stage (see section 3.3) we developed a prototype that was a major extension to the previous stage in terms of HCI and NLP principles. The last iteration (see section 3.4) was done to ensure the interface design and interaction complied with INFOVIS principles. Within the scope of this thesis we have been working on the two redesign phases (sections 3.3 and 3.4).

The ontology mapping has been done in Stage 1 (see section 3.2) and has been presented in [35]; all data parsing, formatting have been done in Stage 2 using Python scripts (see section 3.3). The formatted files were stored offline to reduce data loading time. The NLG component used to derive the abstractive summary is also based on Java and SimpleNLG api and uses the sentence aggregation technique described in [35]. Since the entire front-end application has been developed using Java, and the backend data processing involves Java and Python, our interface is platform independent and free-source. In section 3.1 we describe how the sentences in a conversation are mapped to knowledge concepts in an ontology like speakers, DA types and entities. This mapping presented in Ontology Web Lan-

guage (OWL)/Resource Descriptor Framework (RDF) format (see section 3.6) is the input we used for further data processing. In section 3.6 we briefly discuss how the abstractive summary is generated for a set of selected concepts on the ontology.

We had started with the interface presented in the section 3.2 and had modified it to the fully functional prototype described in section 3.3. We further modified the interface based on INFOVIS visual encoding and interaction principles and the latest design is presented in section 3.4.

3.1 Data Abstraction

Although the goal of our research is to design an interface suitable for analyzing multi-modal conversational data, in this thesis we have been working mainly on the AMI meeting corpus [13]. The meeting conversations in the AMI corpus are structured as a series of 4 meetings the (a) kickoff, (b) conceptual design, (c) detailed design, and (d) evaluation meetings imitating product design cycle. In addition, the meetings have a fixed size group of participants playing very specific roles, like the project manager, the marketing expert, the interface designer and the user interface expert. We have chosen to work on the AMI corpus since it is one of the most frequently used and publicly available corpus in NLP research with different annotational data. By annotations we mean that the human judges have manually labelled the data for the phenomena relevant to the task (summarization in our case) [12]. Currently the interface can be used to analyze a single transcript, but in the future we'll extend it to analyze the whole series of transcripts together.

As described in the Introduction in chapter 1, we use an ontology containing nodes for speakers, dialogue acts, and a list of entities referred to in the conversation to identify more informative sentences or utterances within it. An example for mapping a sentence to the ontology can be found in Figure 3.1.

A: Let's go with a simple chip.

Speaker: A, who is the Project Manager

Entities: simple chip (only one for this example)

Dialog Acts: classified as decision and positive-subj

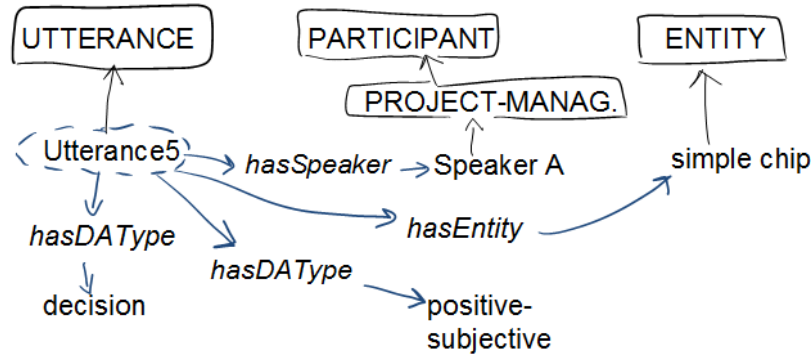


Figure 3.1: Example of mapping a sentence to an ontology

Each conversation thus can be considered as a dataset of sentence items. The attributes of each of these items would be binary variables (possible values ‘yes’ and ‘no’) indicating whether a particular sentence can be mapped to a particular concept on the ontology (DA, speaker and entity). Additionally, the time of utterance or the sequential order of a sentence in the conversation can be considered an ordinal value attribute based on which the dataset is sorted before display. For the above mentioned example item, the attribute ‘ProjectManager’ (say) would have a value ‘yes’ but the other speaker attribute columns would contain a ‘no’.

3.2 Stage 1 Design

The details on an initial interface to aid creating visual structured summaries of conversations can be found in [10]. This very primitive, proof of concept interface relied on mapping the utterances of the conversation into an ontology that then could be used to search the conversation according to the annotation. The ontology in this interface contained nodes for participants of the conversation and for the dialog acts that can be expressed by an utterance or sentence such as a decision, a problem etc. The initial design contained two panels - a panel on the left for displaying the transcript and another on the right to show the ontology (see Fig 3.2). The transcript panel showed the sentences of the conversation one per row, ordered temporally, prefixed by the sentence identifiers. The ontology was presented in a tree-structure allowing multiple node selection using checkboxes juxtaposed to the

node labels. Each node on the ontology was assigned a distinct color. Given the information shown in the two panels, the users could generate visual, structured summaries by selecting nodes in the ontology. As a result, the sentences that could be mapped in the selected nodes would be highlighted using the color assigned to the selected node. If a sentence could be mapped into multiple selected nodes, the highlight color would be a combination of the colors used for the original nodes. This stage was implemented in the GATE system which is primarily suitable for text annotation but falls short as a visualization tool.

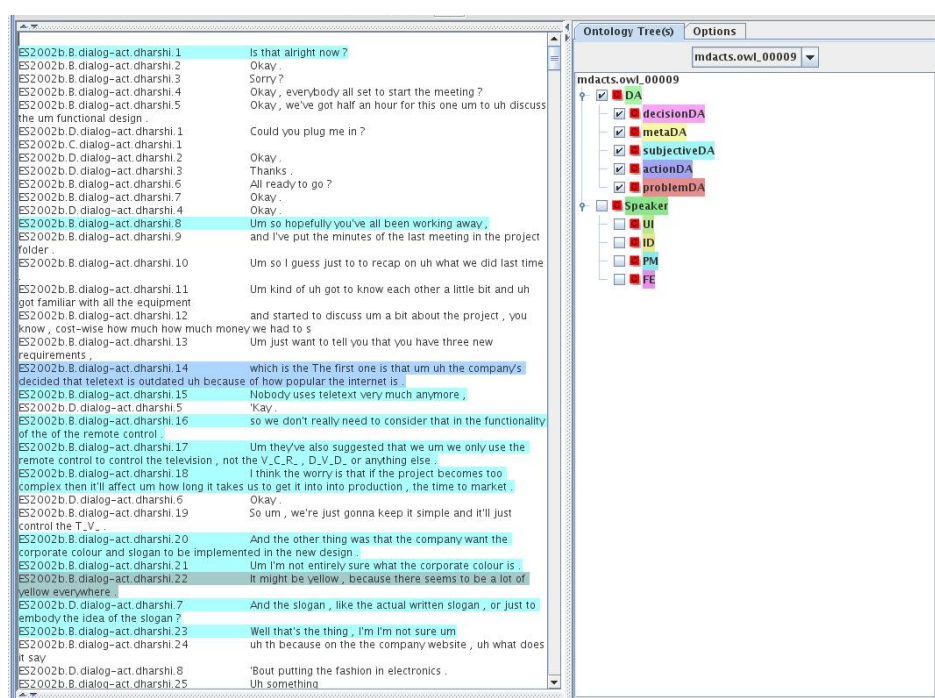


Figure 3.2: Stage 1 design

3.3 Stage 2 Design

The Stage 2 design (see Fig 3.3) addressed several limitations of the initial prototype presented in the previous section. It consisted of three integrated views - the Ontology View, the Transcript View and the Summary View. The details on the display design and interaction design of this stage can be found in [40]. However,

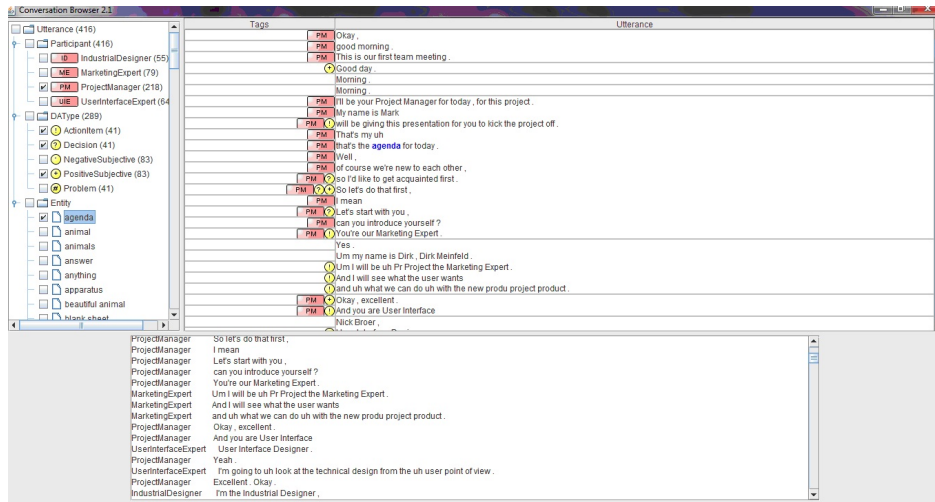


Figure 3.3: Stage 2 design with 3 integrated view - the Ontology View (left), the Transcript View (right), the Summary View (bottom)

we report them here for the sake of completeness. We also discuss the rationale for our redesign in Section 3.3.4.

3.3.1 The Ontology View

The ontology view provided a structured way for the users to explore all the relevant concepts in the conversation and their relations. It contained a tree hierarchy with core nodes Speaker, DAType (Dialogue Act Type), and Entity. Conceptually the top node in the ontology tree represented all the utterances or sentences in the conversation, while any other node represented a subset or subclass of those sentences that satisfied a particular property. For instance, the node ProjectManager (PM) represented all the sentences uttered by the PM, while the node ActionItem represented all the utterances that were classified as containing an action item. The Entity core node, on the other hand, did not represent all the noun phrases detected in the conversation but only the ones deemed important on the basis of their frequency (with mid range document frequency as explained in [33]) and the ones retained after filtering for non-content words and stop words (words like ‘anyone’, ‘okay’ etc.).

As shown in Fig. 3.3 (like [10]), the nodes each had a check box and a label.

Additionally we had included a count within parentheses beside the labels. For leaf nodes, the count indicated how many sentences were mapped to this node and conveyed its relevance for the summary; for non-leaf nodes this was just the sum of all its descendant leaf node count; where a count is an indication of the node's relevance. For Speaker subtree nodes the sets are mutually exclusive and these counts gave a sense of how dominant a role was in this particular meeting. For Entity and DAType subtrees, the sets can be overlapping and the counts at the non-leaf core nodes indicated the extent of overlap. Our interface at this stage displayed the entities in an alphabetical order.

3.3.2 The Transcript View

The Transcript View was designed to allow the users to inspect the whole conversation as well as the mapping of each sentence into the ontology. This view had two columns in Stage 2 - Transcript and Tags. The Transcript column displayed the whole conversation one sentence per row, while keywords and icons for the nodes in the ontology to which each sentence was mapped to were shown in the corresponding Tags column (to the left of the Transcript column), in case of selected nodes under the Speaker and DAType core nodes; or highlighted in the Transcript column, in case of nodes under Entity core node. We had decided to display the entities highlighted in the transcript instead of mentioning them in the Tags column so that the users could inspect them in their actual context. Also, adding a number of long noun phrases to the Tags column would have widened that particular column space making it difficult for the users to inspect both the Tags and the Transcript columns at the same time.

The Transcript View was scrollable both vertically and horizontally which could be used to inspect a sentence in its context i.e. its position in the conversation. A sentence may convey additional information in conjunction with its surrounding sentences. For example, when users inspect the sentence 'That's it, you just put it on the board.' mentioning the entity 'board' in its context, they may decide to include the entity 'pen' for further investigation since the 'it' in the sentence refers to the 'pen' that appeared in a preceding sentence.

3.3.3 The Summary View

The Summary View was a text area where the candidate summary of the conversation appeared for user assessment. The summary was based on sentences selected from the transcript using the criteria set from the ontology tree and at this stage was generated using extraction. The Summary View provided an easier way to assess the conversation overview on a particular information need without scrolling through the whole transcript. To support the users in interpreting the summary in the context of the whole transcript, each sentence in the summary view was prefixed with a keyword indicating the speaker of the sentence.

3.3.4 Redesign Rationale

The capacity of visual channels can be measured by how many level of information they can convey, whether they can be interpreted separately or are automatically merged. One of the major issues with the initial interface was the use of highly saturated colors to highlight sentences because using highly saturated colors may distract the user from the cues provided using other visual channels. A basic design guideline of using color encoding is to use pastel colors when the colored regions are large, such as backgrounds [32]. The color channel loses discriminability after a dozen or so levels, so the solution presented at Stage 1 (see section 3.2) was non-scalable, as a sentence could be tagged with multiple labels and using a combination of the colors of the original nodes becomes perceptually indistinguishable very quickly. If we consider a sentence with just one Speaker tag and one DA type tag there is $4 \times 5 = 20$ possible colors that could be applicable, and human beings can hardly distinguish more than a dozen colors. The actual situation is far worse as a sentence can be classified as multiple DA types simultaneously. Another conceptual problem with the initial solution was using colors to show two different categorical attributes (Speaker, DA type) when the color map for categorical + categorical attributes are perceptually inseparable [32]. The solution we came up with in this stage was, instead of using color, we used tags in a separate column to show the mapping.

We had also included entities mentioned in the conversation to the ontology representation at this stage. Searching the conversation using a particular keyword

can only be done when the user has previous knowledge about the content and want additional information on a particular entity. Representing a list of entities enables the user to perform a more refined search and makes browsing of the conversation easier. In addition, the entities also provide the user with a quick overview of the content of the whole conversation without browsing the transcript. We had also included count within parentheses beside the labels on the ontology tree as information scent [53]. For leaf nodes, the count indicates how many sentences were mapped to this node and imply its relevance for the summary.

To take advantage of visual popout, we had decided to use icons associated with the ontology concepts as in [41], instead of text labels, as representation of tags in the Tags column. We had used pink rectangles for the Speaker type tags and yellow circles for the DA type tags to make the ontology core concepts distinguishable using color and shape channel. A word can have different meanings in different contexts, that's why instead of showing the entity tags in the Tags column, we were highlighting them in bold blue font within the transcript sentences.

We had included gridlines in light gray in the transcript to make the separation of the temporally ordered sentences (one per row) apparent. In the Utterance column, the sentences spoken by a speaker subsequently without intervention from another speaker (known as turns in NLP terms) were grouped using containment within a larger grid box.

The Summary View was a new addition at this stage that worked as a filtered view of the sentences that were mapped to the nodes currently selected on the ontology. Although these sentences can be inspected in the context of the transcript, they may be highly dispersed and the length of the conversation may make it impossible to display them in a satisfactory way within the currently viewable portion of the transcript. The Summary View is linked to the Transcript View as well. Clicking a sentence in the Summary View highlights the corresponding sentence in the Transcript View (along with the two preceding sentences and the two subsequent sentences to make the highlight easier to spot) and also adjusts the viewport on the Transcript view to show the highlighting by auto-scrolling.

The Summary View is an important addition to the interface also because it decouples the task of identifying informative sentences and the task of generating a focused summary for those selected sentences. This makes it possible to choose

either an extractive or an abstractive [35] approach for the generation task. A user after she has inspected the conversation through the mapping to the ontology may wish to generate summaries covering only some aspects of the conversation (which are especially relevant to her current information needs). For instance, she may need a summary of all the *positive* and *negative* comments that were expressed in the conversation about two particular entities (e.g., *new design* and *interface layout*).

3.4 Stage 3 Design

Most of the basic concepts from the Stage 2 design have been preserved in this stage. However, we have modified the interface to be more visually representative of the data using INFOVIS principles (see Fig 3.4). We have improved the data encoding by making efficient use of more visual channels and have provided new ways of interaction. We have also changed the layout of the multiple views to make better use of screen real estate. The details of these modifications and the rationale behind them are enumerated in the following subsections.

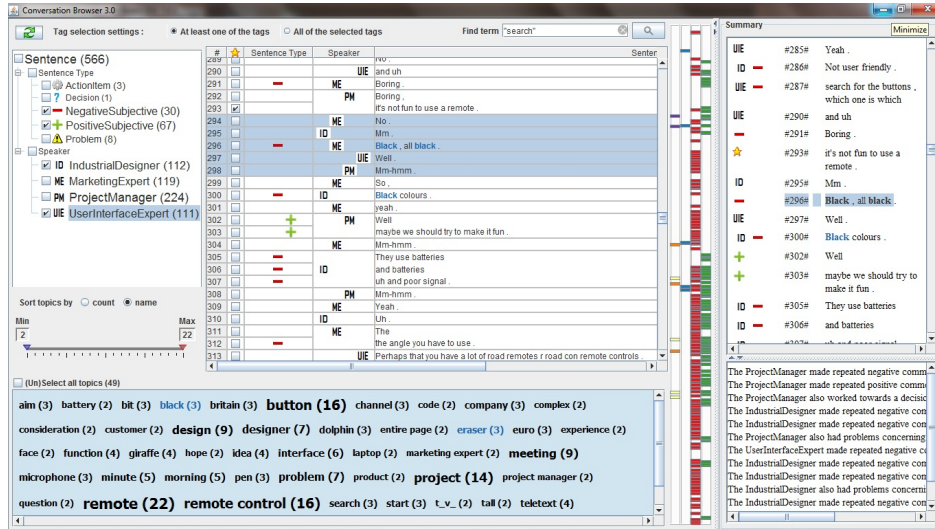


Figure 3.4: Stage 3 design based on InfoVis principles

3.4.1 Display Design

Once the ontology is populated with the participants, DA types and entities of a particular conversation, the transcript of the conversation is displayed ordered temporally. The design of the interface is intended to satisfy two key goals. The first goal is to support the exploration of the conversation through its annotation using speaker, DA and entity concepts in an ontology. This is achieved by allowing the users to select subclasses from the ontology that seem promising to fulfill their particular information needs and by allowing them to inspect the sentences that are associated to those subclasses in the context of the whole transcript. The second goal is to support the generation of focused summaries that cover only aspects of the conversation which are especially relevant to the users. This is achieved by allowing the users to select classes of sentences that they find particularly informative and that should be included in the summary (include verbatim for an extractive summary vs. include their content for an abstractive one).

In this section we discuss in more detail how the achievement of these two key goals is supported by our interface shown in Figure 3.4. Our visual interface consists of four integrated views, the Ontology View (left), the Entity View (bottom), the Transcript View (middle) and the Summary View (right). Our interface does not feature audio-video data streams in addition to transcripts as in Meeting Miner [9] or Ferret [52] because we have designed it to explore and summarize multi-modal conversations in general.

The Ontology View

The Ontology View at Stage 3 is the same in functionality as in Stage 2 except it is now showing only the concepts for Speaker and DA Type nodes (see Fig 3.6). We are displaying the Entity type nodes in a separate view called the Entity View discussed later. The rationale behind this change can be found in the Data Encoding subsection 3.4.2. We are scaling the leaf labels in the Ontology View according to the count to make frequently referred nodes stand out more.

The Entity View

The Entity View (see Fig 3.7) is a textual collage of the list of entities referred to in the conversation represented in a tag cloud format, in a rectangular tag arrangement. The entities are listed in a sequential line-by-line layout. It is possible to select multiple entities simultaneously from the list and the color channel is used to differentiate the set of entities that are selected from the unselected ones. We are using black for the unselected entities, blue for the selected ones and white on mouse hover over an entity tag. The frequency counts for the entities are shown beside the entities within parenthesis and the font size of the labels is made proportional to the count. To make it clear to the user that there is no correlation between the sizes of the labels in the two views (Entity and Ontology) since the scaling function are different we have also set the background color of the Entity View differently from the Ontology View.

The Transcript View

The Transcript View (see Fig 3.8) has a number of tag columns as new additions at this stage. The ‘Sentence’ column displays the whole conversation one sentence per row. Icons for the nodes of type Speaker and DAType core nodes in the ontology, to which each sentence was mapped to, are shown in the corresponding ‘Speaker’ and ‘Sentence Type’ columns (to the left of the ‘Sentence’ column). In case of selected entities, the terms are highlighted in the Sentence column in a bright blue color (refer to the user scenario in section 3.5). We have decided to display the entities highlighted in the transcript instead of mentioning them in a separate tag column so that the users could inspect them in their actual context, as words can have different meanings in different context. The Transcript View also lists the line number at the leftmost column and an override column (with header star icon) that provides a way for the user to mark a particular sentence as important.

The Summary View

We are showing both the extractive and abstractive summaries generated according to currently tagged sentences in the resizable Summary View panel. We have pre-

fixed the summary sentences with tags for the selected ontology concepts that can be applied to the sentence to make the summary view self-contained without looking up the transcript. Also, we have added the line number for the sentences in the conversation in the Transcript view as well as in the Summary view. This provides the user an idea whether the sentences being selected are concentrated at a portion of the conversation indicating a possible topic shift. Providing these types of information is even more important for the abstractive summary approach since for this approach the summary sentences are aggregations of a set of transcript sentences and without the cues the user has to rely completely on the quality of the abstractive summary generation component, which like any other machine learning approach, has a degree of error involved. Details on how the abstractive summary is generated can be found in section 3.6. For abstractive summary sentences, we have provided a list of line numbers for the component transcript sentences after each summary sentence (see Fig 3.5).

3.4.2 Data Encoding

Our interface has been designed to be visually representative of the data using INFOVIS principles (see Fig 3.4). We are using the shape, color, size, and position visual channels for encoding of data attributes or core concepts. The first two channels, shape and color, can be used efficiently for a relatively limited number of levels before they start losing distinguishability. On the other hand, theoretically, size and position can be used to encode a much larger number of levels but they take up screen space which is a limited resource. Although spatial position is the most efficient channel to encode all types of data (quantitative, ordinal or categorical), text itself has an inherent linear order and displaying text in a legible manner puts it up as a contender for screen space. The nature of each of these data channels and their efficiency measured in terms of accuracy, discriminability, separability, and the ability to provide visual popout have been taken into account while deciding the encoding of our data attributes (speaker, DA type, entity etc.).

The count within parenthesis beside the node labels on the ontology tree act as information scents; a higher frequency indicating repeated occurrence of that particular type within the items of the dataset. For a better visual representation

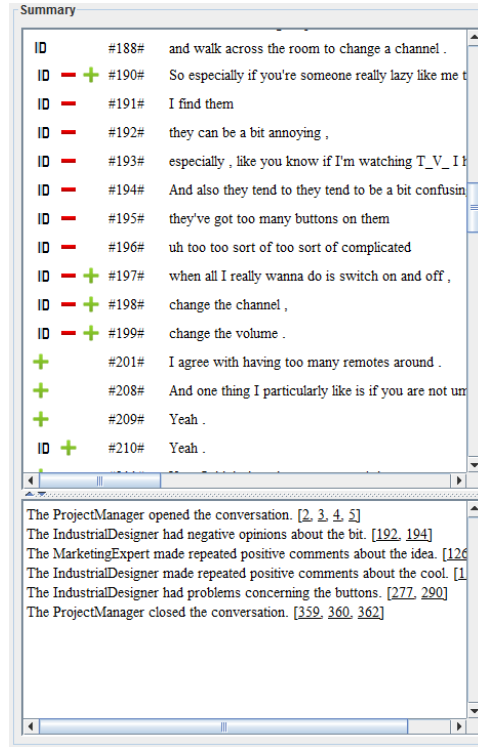


Figure 3.5: Summary View with information scent; the top sub-panel displays an extractive summary while the bottom one shows an abstractive summary for the tagged sentences

of this information scent, in this stage, we have scaled the font size of the labels (see Fig 3.6) i.e. we are using a larger font for nodes with larger counts to make them stand out more. However, the frequency distribution for the three core nodes DA type, Speaker and Entity are not the same. Entities in a conversation rarely appear more than 5 or 6 times. On the other hand, since each of the sentences can be mapped to a speaker, for a sizeable conversation (around 1000 sentences) each of the speaker nodes would have hundreds of sentences mapped to it. The DA type leaf nodes have a distribution similar to speakers. To keep the labels legible we had to impose a minimum and maximum font size for the scaling. Given these boundary conditions, using a linear function to scale the fonts for the leaf nodes was not possible as it would assign the maximum font size for most of the nodes

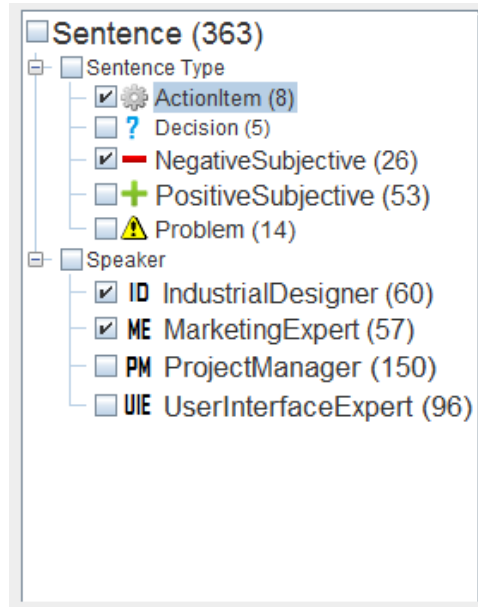


Figure 3.6: Ontology View concept encoding using size, shape and color channels

under the Speaker and DA type core nodes. Using a logarithmic function would not provide a satisfactory solution since it would not reflect the variation in the frequency of the entities. As a solution to this completely different distribution of nodes under the core nodes, we are displaying the entities in a tag cloud format (see Fig 3.7) separate from the Ontology View. This has allowed us to use two different scaling functions for the labels in the two different views; a logarithmic scaling function for the Ontology View containing nodes for the DA type and speaker concepts and a linear scaling function for the Entity View representing the entity concepts.

button (15) remote (13) t_v_ (10) bit (9) project (9) television (9) animal (8) idea (7)
meeting (6) alima (5) battery (5) **remote control (5)** video recorder (5) channel (4) discussion (4)
favourite animal (4) kay (4) euro (3) **individual work (3)** california (2) cool (2) feature (2) home (2) introduce (2)
mind (2) project manager (2) room (2) **snake (2) talk (2)** thanks (2) user interface designer (2) water (2)

Figure 3.7: Entity View using size and color for encoding and position as sort order

The DA types in our interface is a fixed size set (size is 5). Although this set could be extended if we used other supervised classifiers trained on annotated data (see section 3.1), for the extant of our project the list is unchangeable, making it possible to use shape and color channels. There is a strong domain convention to use red for negative sentiments and green for positive sentiments, which we have followed to design icons for PositiveSubjective and NegativeSubjective DA types. We have used luminance to counter the limitation of using red-green hues for color-deficient users and have made the other DA type icons of distinctly different colors. We have also redundantly used the shape channel for the DA type concepts. So, we are using a green '+' or a red '-' shaped icon for PositiveSubjective and NegativeSubjective nodes respectively. For the other DA type nodes, we have used the shape of the most common icon found when we googled using the keywords (see Fig 3.6).

Although the speakers in the AMI meeting conversations are fixed, this is not the case with email and blog discussions where the group of participants is of variable size. That is why shape and color are not scalable solutions for the speaker nodes. Also, there is no intuitive mapping from a person's name to shape or color and users of the interface are interested in knowing exactly which participant is saying what rather than just identifying a change of speaker. As such, we have decided to use the abbreviations of the speaker names as representative glyphs for the concepts (see Fig 3.6).

The speaker is incorporated as a turn parameter which has two-fold advantages (see Fig 3.8). Firstly, the icon for a particular speaker appears only for the middle sentence of a turn, this reduces the time the user would have spent in verifying who the speaker is if it appeared for every line of text. Secondly, although speaker is still a filtering criteria selectable from the Ontology View, we are displaying the speaker icons all the time in the Transcript View to help users maintain orientation. Separating the columns reinforce the behavioural differences between the DA type and speaker concepts.

We are also redundantly coding these attributes using spatial positioning in the Transcript View. In the 'Sentence Type' and the 'Speaker' columns, icons for specific leaf nodes of these categories appear in specific locations. For example, in figure 3.8, the 'Sentence Type' can be seen as consisting of 5 subcolumns (without

#	★	Sentence Type	Speaker	Sentence
257	<input type="checkbox"/>	+	PM	Mm .
258	<input type="checkbox"/>	? + ⚠	UIE	and if um if you'd save the more complicated functions maybe for separate
259	<input type="checkbox"/>		ME	Mm-hmm .
260	<input type="checkbox"/>		PM	Okay ,
261	<input type="checkbox"/>	? + ⚠		so maybe have like one remote that has the main functions on , off , channel
262	<input type="checkbox"/>		UIE	Um .
263	<input type="checkbox"/>		PM	Because that is one thing that um remotes tend to have buttons that the
264	<input type="checkbox"/>	+	UIE	Yeah .
265	<input type="checkbox"/>			Mm .
266	<input type="checkbox"/>		PM	So like you have to have them somewhere ,
267	<input type="checkbox"/>		ID	Yeah .
268	<input type="checkbox"/>		PM	'cause you're gonna m need those special functions occasionally .
269	<input checked="" type="checkbox"/>		UIE	Right .
270	<input type="checkbox"/>		PM	Um but not necessarily on the m the normal remote .
271	<input type="checkbox"/>		ID	Can I ask ,
272	<input type="checkbox"/>	⚠		are we designing a remote control for a television only
273	<input type="checkbox"/>		PM	Good question .
274	<input type="checkbox"/>		ID	um , and if if this device is just to be used for the television
275	<input type="checkbox"/>	⚠		would we even technically be possible to include video recorder functions on
276	<input type="checkbox"/>		PM	I don't know that yet .
277	<input type="checkbox"/>	⚠	ID	Um or should we just stick to just stick to having television television related
278	<input type="checkbox"/>		UIE	Mm .
279	<input type="checkbox"/>		PM	It's a good question . Um .
280	<input type="checkbox"/>	⚠		I'll look into that .
281	<input type="checkbox"/>		ME	Mm-hmm hmm .
282	<input type="checkbox"/>		PM	If I can .

Figure 3.8: Transcript View with separate columns for Speaker and DA type nodes and Speaker incorporated as turn parameter

any visible borders separating them), where the green '+' icon for 'PositiveSubjjective' node is applied to the 4th subcolumn, in the order of its appearance on the ontology tree hierarchy under the 'Sentence Type' core node. The 'Speaker' type nodes also appear in specific location under the 'Speaker' column. This solution for the 'Speaker' tag is applicable for the AMI meeting dataset only and cannot be extended to the email and blog conversations due to their varying list of participants (the number of participants in blog conversation can even reach upto hundreds).

3.4.3 Interaction Design

When the Transcript View is generated for a conversation, the Sentence Type column is initially empty and all the nodes on the ontology tree in the Ontology View

are shown and are de-selected. The list of entities in the Entity View gives the user an idea of the conversation content without requiring them to browse the whole transcript. If for a particular conversation the ontology is too large, the user can expand/minimize nodes she is/is-not interested in, as in any standard outline based interface. Similarly, if the list of entities is too long, she can use the range slider for the Entity View to shorten it. Once the user selects a node (or de-selects an already selected node) on the ontology tree, the keyword or icon associated with that node appears in (or disappears from) the Sentence Type column of all the rows that contain sentences that can be mapped to that particular node, in case of Speaker nodes, the icons are shown all the time but selecting a Speaker type node on the Ontology View changes the filtering criteria for the summary. Once the user has selected the nodes of interest from the ontology tree and the Entity View tag cloud, she can scroll through the transcript view and inspect sentences that appear to be promising for generating a focused summary.

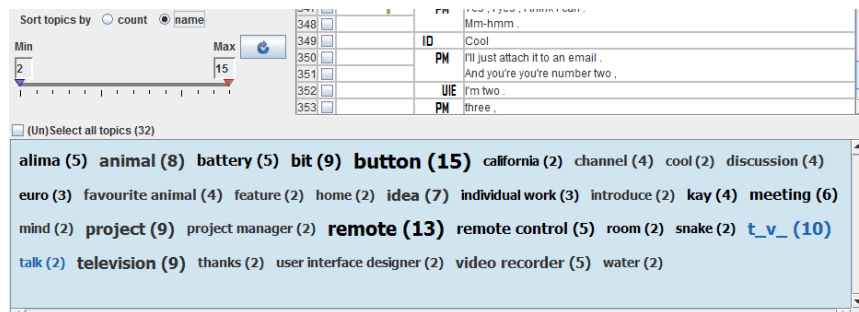


Figure 3.9: Entities sorted by name using control

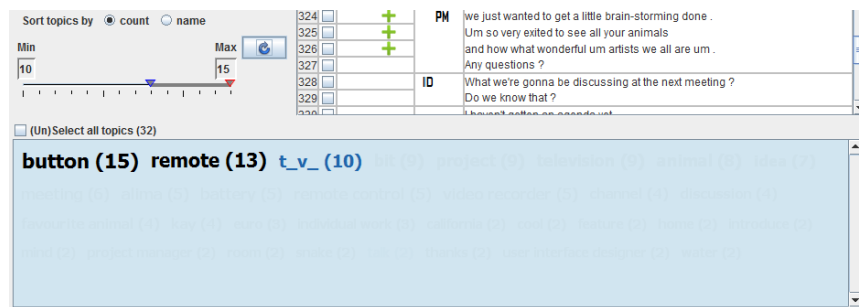


Figure 3.10: Using range slider to shorten list in Entity View

The Entity View is a useful entry point to the conversation providing a list of precompiled keywords. However, the long list of entities impose cognitive load on the user. To enable users to concentrate on a subset of the entities, we have listed them in a descending order of the frequency count and have provided a range slider where the user can specify a minimum and a maximum count for the list of entities. This fades out the entities falling outside the range selected and thus narrow down the search scope. Also, we have provided a control to change the sort order of the entities to alphabetical listing (an important aspect of text data). This is an example of interaction to change spatial position of the elements according to sort order (see Fig. 3.7, Fig. 3.9 and Fig. 3.10). This sort order might be handy to the user if she was looking for a specific topic but did not know its frequency of occurrence. Sorting the entities in lexical order would allow her to skip to the part of the list where the topic is expected to appear (for example, an entity starting with an ‘m’ would appear somewhere in the middle close to entities starting with ‘l’ and ‘n’) and she would be able to make an exhaustive search. If the Entity View were listed in the order of the frequency she might have to go through the whole list to find a particular entity and in case of a long list she could easily miss the one she was looking for.

By selecting a few entities out of the list the users can satisfy particular information needs on the direction the conversation took regarding those particular entities. For instance, a user may be interested in all the comments made by the ProjectManager on the ‘board’ and whether these comments were positive or negative. To achieve this goal, the user would select the node ‘board’ under the Entity View, the concept ‘ProjectManager’ under the Speaker core node and ‘PositiveSubjective’ and ‘NegativeSubjective’ nodes under the Sentence Type node. As shown in figure 3.5, this would display the representative icons ‘+’ and ‘-’ in the Sentence Type column of sentences that map to each of these nodes and would also highlight every occurrence of the word ‘board’, providing the user scope for closer inspection by scrolling through the transcript.

The summary view works as a filtered view of the Transcript View showing all the sentences that are tagged according to the nodes currently selected on the ontology from the Ontology View and/or the Entity View. The result set of tagged sentences may be narrowed down by making the Tag Selection Settings set to ‘All

of the selected tags' or may be widened by setting it to 'At least one of the selected tags'. The user can reach a good sized result set by selecting or deselecting ontology concepts and toggling the Tag Selection Settings; which is then presented in the Summary View. The Summary View is linked to the Transcript View where the user may navigate by clicking a sentence. As a result, the corresponding sentence(s) in the Transcript View is(are) highlighted (along with the two preceding sentences and the two subsequent sentences to make the highlight easier to spot) and the viewport on the Transcript View is adjusted to show the highlighting by auto-scrolling.

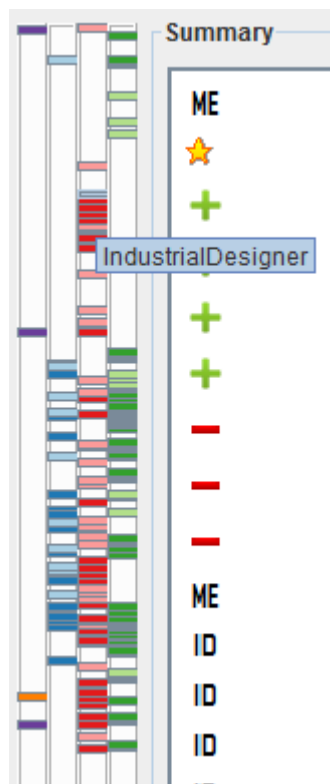


Figure 3.11: Marker bar with color encoding core concept and tooltip texts

Despite the visual cues provided as tags for the transcript sentences and the link between the summary view and the transcript view, the user still has to spend considerable time scrolling through the transcript since generally inspection of the

context sentences is mandatory to fully understand the relevance of the tagged sentence; and since sometimes the tagged sentences are widely spread out throughout the conversation length. We have added marker bars (see Fig 3.11) as a new mechanism for interacting with the Transcript View to reduce this scroll time. We have decided to use four parallel marker bars instead of just one to reduce visual clutter. There is one marker bar dedicated to each of the ontology core concepts (speaker, DA type and entity) and along each of these bars the most recently added tag of the type is shown in a highly saturated color while the older tags are shown in less saturated shades of the same color (blue for entity, red for participant and green for DA type). We have chosen the three basic colors red, blue and green for the three concept types since the best color choice for categorical map are the fully saturated colors that are also easily nameable, such as red, blue, green, yellow, white, and black. When more colors are required, the next best set of good choices are orange, brown, pink, magenta, and purple [32, 48]. We have chosen the two shade encoding assuming that users would be adding a single concept (or a relatively small number of concepts) at a time as filtering criteria and would want to inspect the effect of the latest addition further before proceeding to adjusting the filtering criteria to satisfy her information needs. There is a fourth marker bar along which (orange colored) markers appears for the sentences the user identifies as important (despite not being tagged). This allows the user to easily find those sentences later to re-inspect them. The mapping from the Summary View to the Transcript View (for abstractive summary) is also shown as (purple colored) markers along this bar. Although the markers on the different bars do not interact with each other, the bars are placed very close to each other; so, we have used a Colorbrewer qualitative palette [3] to choose the colors of the markers. The Colorbrewer application provides coloring advice for map design and is a diagnostic tool for evaluating individual color schemes in terms of robustness. We have also set the background of the marker bars to a very light shade of gray so that the effect of the color coding is not interfered with. Due to scalability issues, only the core level concepts could be color coded. To distinguish leaf level concepts we are using the tooltip text of the markers. Whenever the user makes the mouse pointer hover over a particular marker, the corresponding concept is shown as a tooltip text. This can be considered as rudimentary details on demand interaction where as the dis-

tribution of markers along the bars act as an overview of the document from the perspective of the concept's occurrence along the conversation length. Clicking on a marker auto-scrolls the transcript and makes the part of the transcript corresponding to the tag associated with that marker visible to the user (using highlighting), this is an instance of change of viewport type of interaction [32].

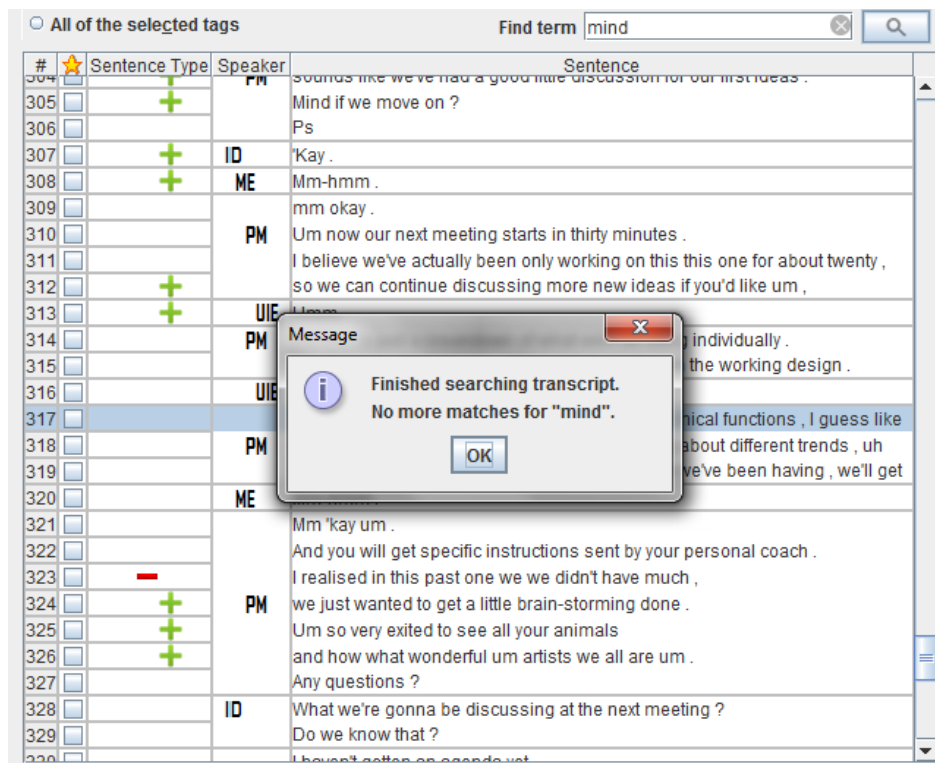


Figure 3.12: Search box to look up keyword

The Keyword Search Box can be used to investigate the sentences in the transcript referring to a search term, one at a time, until the end of the transcript has been reached. The user can use the Search button or the enter key on the keyboard to cycle through the occurrences of the keyword and when there are no more occurrences of the search term, a popup message box is shown to the user indicating that (see Fig 3.12). Whenever the user searches for a new keyword, if it exists, a set of cream colored markers is placed on the first marker bar corresponding to the position of the sentences referring to the word or phrase. Clicking on these markers

is an alternate way of inspecting the relevant sentences.

We also provided a Reset Interface button at the top left corner of the interface which a user can employ to deselect all currently selected nodes (of type speaker, DA type and entity) and all override markers for sentences in a single click. This button can be used to reset the interface to the initial state after loading the conversation. This feature can be helpful if the user wanted to start from a clean slate when moving on from one task to another or when searching for a good filtering criterion on a trial-and-error basis.

3.4.4 Layout Design

We have attempted a layout of the multiple views in terms of use of screen real estate and probable usage pattern of each of the views. We estimated the probable usage frequencies based on feedback provided by the participants at the pilot study (see section 4.1) that we conducted after Stage 2 design (see section 3.3). We have also tried to place the controls and interaction mechanisms for a particular view close to it.

The filtering controls like the Ontology View and the Entity View have been placed towards the left side of the interface similar to the placement of menu on webpages and navigator on standard editors etc. We have placed the Summary View to the right side of the interface to emphasize its role as a result of the interaction (even though the summaries can be used as interaction mechanisms themselves). This places the Transcript View at the centre of the layout in keeping with the notion of centrality of the conversation transcript to the task at hand.

We have placed the array of marker bars beside the Transcript View to emphasize their role as an interaction mechanism for browsing the transcript. We have also made the bars use the full length of the screen space available since even after providing separate bars for the ontology concepts there is still a considerable amount of clutter induced by overlapping markers due to space contention. When allocating the positions for the parallel marker bars dedicated to different concept types, we have taken into consideration the probable usage of the particular core concepts (see section 4.1). Since users find the Entity concept most important for identifying informative sentences the bar for it is closest to the Transcript View,

after that we have placed the Speaker marker bar and the DA type marker bar. We have given the fourth marker bar for the override and summary sentence map highest priority and have set it right beside the Transcript View for two reasons; due to the small number of markers that are placed along it, it is easy for the user to overlook it if it is placed farther amidst the cluttering of colors on the other three bars. Also, the override control is a direct reflection of the user's current information need which is what we are striving to satisfy using this interface. Similarly, the range slider and sorting controls for the Entity View are placed close to it. The generic search box is placed close to the transcript. We have positioned the reset interface button and the tag selection settings at the top to emphasize their effect encompassing both the Ontology View and Entity View.

We are using a very simple dynamic layout to put the screen real estate to better use according to the user's preference. The summary view can be kept completely collapsed while the user is focusing on conversation analysis using marker bar or keyword searchbox widgets. If the user wishes to use the Extractive Summary View as the primary means of navigation she can maximize the number of lines shown without scrolling by minimizing the area dedicated to Abstractive Summary View. Once the user has generated her focused summary with the help of the interface using the concepts in the conversation, she can maximize the Summary View to inspect the summary further with minimal scrolling.

3.5 Scenario of Use

One of the uses of the interface we have presented here would be as an analytical tool by the NLP experts. Almost all NLP applications provide results in a flat text format which is nearly incomprehensible through manual inspection; and as such, statistical measures are used to verify the performance of these applications. However, statistical measures are not exhaustive and some patterns are easier to identify using human perception. We are relating such an anecdote from the early development phases of the interface. Right after adding in the count for the ontology concepts (see Fig. 3.6), it became clear that the counts for the 'PositiveSubjective' and the 'NegativeSubjective' nodes seemed to be highly correlated. To verify whether that was a coincidence or not, we loaded different conversational test data

on the interface and also checked one of the input files manually to make sure the problem was not due to a programming ‘bug’ on the interface end. It turned out that the error was introduced during a recent modification to the background ontology mapping application. It became apparent that the interface could be a useful verification tool as NLP applications are highly dependent on annotating data using different aspects. An NLP expert could easily extend the ontology based on additional annotation concepts and use the interface to verify whether the algorithm is behaving in the way it is expected to.

The summary generated interactively using the interface could also be helpful for satisfying specific queries of a user on a particular conversation thread. Consider a scenario where an employee has recently joined a product designing company. Automatic meeting abstracts would allow this new personnel to prepare for an upcoming meeting or review the decisions made by a previous group. This person could be specifically looking for the commonly-used functions of the remote control that the group was designing. Using our interface this new employee could easily browse the transcripts (may be generated by Automatic Sound Recognition (ASR) or as a part of meeting minutes) and find out what was the final decision, whether there was any alternatives the team considered, and what was the reasoning for and against any alternatives.

Given the above task description, the user, aided by our interface, might first skim through the tag cloud of the Entity View and select entities like ‘remote control’ and other entities like ‘button’ or ‘lcd’ that seem related to remote control designs (see Fig. 3.7). This would highlight those entities in the transcript and markers along the marker bar would appear (see Fig. 3.4). At this point the user could employ two different approaches to inspect the sentences further; she could click on each marker (see Fig. 3.11) and read the corresponding portion of the transcript, or she could look at the filtered sentences in the summary view (see Fig. 3.5) and make informed choice on which sentences to inspect further in the transcript. The link between the summary view and the transcript view would reduce the scroll-time if she decided to use this approach. Alternatively, when using the marker bar as the main form of interaction she could look at the distribution of the selected entities along the bar and first try to inspect the part of the conversation where the relevant entity keywords appear close to each other i.e. where the mark-

ers for the phrases ‘remote control’ and ‘button’ are concentrated. The tooltip text provided for the markers would help to identify such regions.

To narrow down search scope for the alternatives considered and the reasoning behind them the user might want to select the ‘PositiveSubjective’, or the ‘NegativeSubjective’ on the ontology view. Further, the user might even employ her real world knowledge that ultimately a decision would be made by the ‘ProjectManager’ and concentrate on ‘Decision’ type sentences by that particular speaker (see Fig. 3.6). To find out what other topics were highly discussed, the user might want to use the range slider provided for the entity cloud and set it to retain entities with higher counts in the conversation (see Fig. 3.10). This would significantly reduce the number of elements shown on the entity cloud and enable the user to concentrate on a much smaller set of entities and try to correlate them with the task on hand. While looking through the transcript she might remember she had seen the word ‘battery’ being mentioned in the conversation at some point and that might be relevant to the power source feature of the ‘remote control’. She could then type in the phrase ‘battery’ in the search box provided for the transcript view to look at all occurrences of the term, one at a time in the temporal order they appear (see Fig. 3.12). On the other hand, she could use the ‘Sort by name’ control for the Entity View and check whether the term ‘battery’ is one of the keywords. The alphabetical ordering of the entities in this mode would make the searching task faster and exhaustive for the user (see Fig. 3.9). If ‘battery’ does appear on the Entity View (is identified correctly by our background application) the user could then select it and resort to further inspection using the marker bar.

3.6 Abstractive Summary Generation

Our browsing and summarization method relies on mapping the sentences in a conversation to an ontology containing three core upper-level classes: Participant, Dialog Act (DA) types and Entities [33]. For our AMI meeting scenarios in which people discuss the design of a new remote control, the Participant class consists of four subclasses ProjectManager (PM), IndustrialDesigner (ID), UserInterface-Expert (UIE) and MarketingExpert (ME). The DA-type class, on the other hand, contains subclasses decisions, actions, problems, positive subjective and negative

subjective sentences. The Entities are noun phrases referred to in the conversation with mid-range (10%-90%) document frequency. Our classifiers are designed for identifying five subclasses of the DA-type class but we could easily include additional classifiers to identify other types of dialog acts according to the information need. We used a feature set related to generic conversational structure, which include sentence length, sentence position in the conversation and in the current turn, pause-style features, lexical cohesion, centroid scores, and features that measure how terms cluster between conversation participants and conversation turns; so, it can be extended to multi-modal conversation data. We have also used sentence level features like word pairs, POS pairs, character trigrams etc. for the classifiers.

Our ontology is at first populated with the instance data for a given conversation. For the AMI meeting corpus, a particular conversation consists of utterances like ‘so I’d like to get acquainted first.’, which have the following format in the ontology:

```
<Utterance rdf:about="#TS3012a.A.dialog-act.vkaraisk.12">
<rdf:type rdf:resource="#owl:Thing"/>
<hasSpeaker rdf:resource="#ProjectManager"/>
<hasDAType rdf:resource="#Decision"/>
<begTime>18.61</begTime>
<endTime>20.49</endTime>
</Utterance>
```

The above utterance is a decision type statement made by the ProjectManager at the meeting. The beginning time of utterance is used to temporally order the whole conversation in the Transcript View and the unique identifier of the Utterance object is used to match the utterance with the actual sentence being said and thus any relevant entities.

We now briefly describe how the abstractive summaries are generated. More details can be found in [33, 35]. The abstractive summary is generated by first combining utterances about pairs of Participants and Entities that repeatedly co-occur into messages. Then the most informative messages are selected by using a optimization function combining sentences/utterances and messages subject to 3 constraints (a length constraint and 2 constraints tying messages and sentences together). The messages relevant to interactively selected utterances are selected

to be shown in the Summary View (see 3.4) bottom panel. An NLG component is used to generate a summary sentence corresponding to a message. The optimization function to sift more informative messages is as follows:

$$\text{maximize}(1 - \lambda) * \sum_i w_i s_i + \lambda * \sum_j u_j m_j$$

Here w_i is the sum of posterior probabilities for the ontology-mapping classifiers for sentence i , s_i is a binary variable indicating whether sentence i is selected, u_j is the number of sentences contained by the message j and m_j is a binary variable indicating whether message j is selected.

Here, a sentence can only be selected if it occurs in a message that has been selected, this constraint can be expressed by the formula below:

$$\sum_j m_j o_{ij} \geq s_i \quad \forall_i$$

where o_{ij} indicates the occurrence of sentence i in message j . Moreover, a message can only be selected if all of its sentences have also been selected. This constraint can be expressed by the formula below:

$$m_j o_{ij} \leq s_i \quad \forall_{ij}$$

The length constraints for the resultant summary can be formulized as below, where l_i is the length of sentence i , L is the desired summary length, and k indicates a threshold for the number of messages selected:

$$\sum_i l_i s_i < L \text{ and } \sum_j m_j \leq k$$

Chapter 4

Evaluation

We had conducted a pilot study after completing the first redesign (see section 3.3) and at the beginning of the second redesign (see section 3.4) to assess which features of the interface are deemed more useful by the study participants and thus demand more consideration during redesign. The details of this pilot study is given in section 4.1. We evaluated our interface in a formal user study after completing the second phase of redesign and collected feedback on the usability of the interface by administering a questionnaire at the end of the study. We also gathered suggestions on how to improve the interface in the future. The user study details can be found in section 4.2.

4.1 Pilot Study

We conducted the pilot study at the beginning of the Stage 3 Design (see section 3.4) after splitting the display of the ontology concepts between the Ontology View and the Entity View but before redesigning the data encoding (presented in section 3.4.2) and the layout (presented in section 3.4.4). We performed a comparative analysis of two conditions i.e. two versions of the interface (one version had some of the features turned off while the other had all available features on). The details of the pilot study are provided in the following subsections.

4.1.1 Participants

We recruited twelve participants through Facebook; six of them male and six of them female. We assigned the same number of male and female participants per condition to ensure there was no gender bias involved between the two groups of participants. We also tried to keep the pool of participants comparable for the two conditions in terms of English proficiency level. All of the participants were graduate students at a North American university. We assumed there will not be an effect on the results due to differences in comfort level of using computers since they were all students in the departments of Computer Science or Electrical and Computer Engineering. The participants were paid a compensation of \$10 for the approximately one hour long study.

4.1.2 Experimental Setup

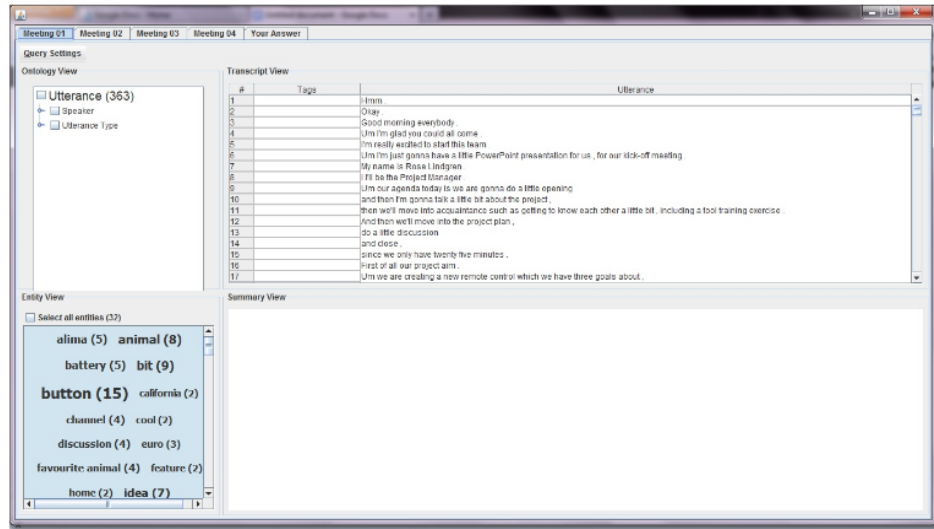


Figure 4.1: Test interface with all features enabled

We conducted a between subjects pilot study on two conditions (1) a test interface with an Ontology View, an Entity View, a Transcript View and a Summary View with only abstractive summary (see Fig. 4.1), and (2) a baseline interface without the DA type annotations and the Summary View (see Fig. 4.2). We kept

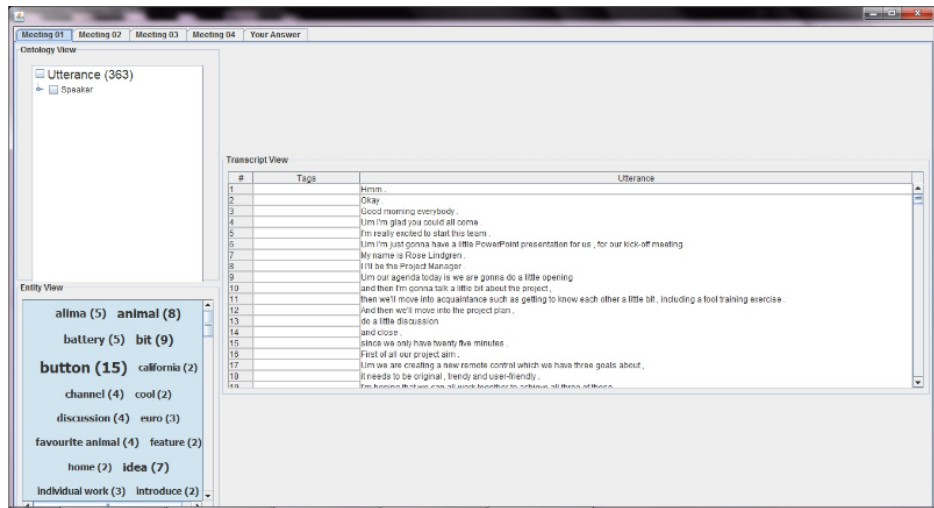


Figure 4.2: Baseline interface with DA type nodes on the Ontology View and the Summary View disabled

the Transcript View the same size in the two conditions since we wanted the same number of transcript lines to be viewable so that the scroll time was comparable for the two conditions.

We used the same set of instructions for both the conditions; the complete task instructions can be found in appendix A.6. The participants were asked to take as much time as they deemed necessary to get used to the interface loaded with the AMI meeting series IS1003. After that, the users were asked to browse the 4 meetings in the AMI ES2008 series where a product design group discusses the design of a new remote control. The meeting conversations in the series were displayed on separate tabs on the interface. We displayed different series of meetings for the practice session and the experiment session to make sure that the participant did not accidentally stumble upon information that can be used to answer the task set in the experiment session while getting used to the interface during the practice session. Details on the ES2008 series of conversations can be found in section 4.2.2. During the experiment, we asked the participants what was the final decision made on separating the commonly-used functions of the remote control from the rarely-used functions of the remote control. We requested them to write a short summary (1-2 paragraphs) about any alternatives considered and the rationale be-

hind accepting or rejecting them. During the session, we automatically logged all interactional behaviour of the participants with the interface like mouse clicks etc.

4.1.3 Marking Scheme

We recruited a native English speaking graduate student as the judge to evaluate the response of the participants. We provided the judge a set of the transcripts and the gold-standard meeting summaries of ES2008 meetings compiled by the AMIDA project group. The gold-standard summaries included 26 items in total and the judge was requested to assess the number of items each participant correctly identified. We also requested the judge to calculate the precision and recall for the response of each participant. The summary of the results is presented in table 4.1.

4.1.4 Results

We summarize the results from the pilot study in table 4.1. The two groups of participants for the two conditions identified the same number of items in total resulting in an average of 5.67 items identified correctly per person. A better precision was achieved by the baseline interface group compared to the test interface group. However, the usage of the Entity View, the Ontology View and the Summary View showed that the use of entities, participants and DA type nodes went down when the Summary View was available in the test interface. Based on these results, we concluded that a summary is capable of serving as an alternative entry point to the conversation.

Condition	Avg. items correctly identified	Precision	Recall	Avg. # of times Summary View used	Avg. # of times Entity View used	Avg. # of times Ontology View used
Test	5.67	0.588	0.218	6	4.33	3.33
Baseline	5.67	0.624	0.218	-	7.17	5.33

Table 4.1: Summary statistics for the pilot study

From the post-questionnaire we administered and the open discussions we held with the participants, we concluded that the Entity View was considered the most

useful component of the interface to find informative sentences. That is why we decided to allocate more space to this view during the second redesign and included controls to aid users employ it more effectively (refer to section 3.4).

Although the participants employed the abstractive summary (when provided in the test interface), they commented on the lack of context and the generic nature of the abstractive summary sentences. Based on this feedback, we decided to include both the abstractive and the extractive summaries for the final user study and leave it upto the user to decide which type of summary to use (if make use of a summary at all).

While judging the participants' answers, we came upon a set of items that were being mentioned repeatedly by the participants (at least four out of our twelve recruitments mentioned 5 such item) but were missing on the gold-standard list. For example, the kinetic power option of the remote control was mentioned by six of our participants (which is half of the number we recruited) but was not mentioned in the gold-standard list. Based on these findings we decided to recompile the gold-standard list of items to make it more complete.

Another problem we faced during calculating the precision of the participants' responses was the uncertainty involving which gold-standard items the users' responses best matched. This issue had arisen because the participants were asked to write a short summary in their own words which could be interpreted in different ways. Also, since there were a number of gold-standard items that were related to a particular topic and hence had significant overlap, it was hard to do the mapping to specific items. That is why we decided to make our final user study queries more specific and to recruit two judges instead of one to counter these types of subjectivity. We also decided to ask the participants for evidence that they had reached the point of the conversation where the answer could be found.

During the pilot study, one participant had extreme difficulty grasping the idea of how the knowledge concepts (like entities) related to a conversation could be used as entry points to it and we had to restart the experiment after providing a brief explanation. This participant suggested that assigning sample tasks during the practice session that were designed to make use of different components of the interface would have helped her understand the workings of different features better. Based on this feedback we designed our user study to have 3 sessions (refer

to section 4.2.2) - (a) a tutorial, (b) a practice and (c) an experiment session with carefully designed tasks that could be completed using different combinations of the components. The first two sessions were designed to ensure that the participant had acquired a working understanding of how the knowledge concepts like speaker, DA type, entities etc. can be used as entry points to the conversation, before moving on to the experiment session.

The groups of participants of the two conditions had identified the same number of items correctly in total (refer to table 4.1) which indicated that turning off the DA type tagging and the abstractive summary in the baseline had minimal effect on a participant's performance i.e. the two interfaces (baseline and test) were too similar to affect the performance of the participants. Since our final interface has a number of integrated views and alternate ways of navigating through the transcript, we decided not to do a comparative analysis in our final user study. Without a statistically large enough usage data for each of the components, it would have been hard to choose a set of features that would have a significant effect on the performance.

4.2 User Study

We conducted a formal user study at the end of Stage 3 Design (see section 3.4) to evaluate our interface. Unlike the pilot study presented in the previous section, this study was not a comparative analysis of two interfaces. Instead we concentrated on gathering the usage data of different views and components of the interface with an objective to analyze whether there were any correlation between the usage data and the score achieved by the participants. The details of the use study can be found in the following subsections.

4.2.1 Participants

We recruited 30 participants through Facebook and email to evaluate our interface. The participants were compensated at a standard rate for the approximately 2 hours they spent on the study including filling up forms and questionnaires. We also offered a prize for the top three scorers at the experiment to encourage people to get engaged in the task assigned.

The subjects ranged in age from 20 to 32 with an average age of 25.3 and median age of 25. Out of the thirty participants twenty were male and ten were female. Most of the subjects were students at a North American university; four of them were pursuing an undergraduate degree and twenty-one of them were graduate students. The five remaining participants held a Bachelor's degree but were not students at the time of the study. Five of the participants were native English speakers. The remaining participants spoke English as a second language and all but two participants were actively pursuing study or had completed a degree at a North American university. These two participants (participants 5 and 9) were excluded from further analysis of the results since they performed well below the average. The rationale behind treating them as outliers is discussed in section 5.1.

We asked the participants to self-assess their comfort level using computers on a scale ranging 1 to 10 (where 1 meant they rarely used computers and 10 meant they could be considered expert computer users) and the median value was 9 out of a range of values from 3 to 10. The participants reported spending on average 7.5 hours daily on computers. Thirteen of the participants were enrolled in or had attained a degree in Computer Science related fields, six never took a computer course and the rest were familiar with basic programming concepts. Eight subjects reported having familiarity (rated 6 or more in a scale of 1 to 10) with visualizing large scale data. Nine of the participants had corrected vision and one reported having occasional blurred vision. All of them reported their vision deficiency to be non-hindering. We performed a series of t-tests to find out whether these factors affected a participant's achieved score. The results were not statistically significant at 0.05 significance level. The detailed results of the tests can be found in section 5.3.

4.2.2 Experimental Setup

At the beginning of the session, we asked the users to fill up a pre-questionnaire to gather some background information (refer to appendix A.2). A summary of the collected information can be found in the section 4.2.1.

In the study the participants were asked to use the visual interface to browse a human conversation to answer questions about the discussions that took place

during the length of the conversation. The study was designed to get the users acquainted to the features of the interface using sample conversations before assigning the actual experimental tasks. The user study was conducted in 3 stages (see appendix A.3) : a) tutorial session, b) practice session, and c) experiment session.

In the tutorial session, the experimenter introduced the interface to the user and explained how the different components or views could be used to answer particular questions about a very simple conversation between two participants and consisting of 50 sentences. At the end of this session, the user was allowed to take as much time as she deemed necessary to get comfortable using the interface. In the practice session, the user worked on a slightly longer conversation of 100 sentences between 2 participants. The user was assigned two tasks based on the conversation and was encouraged to work on her own but was also notified that she could ask the experimenter for help, if necessary. At the end of this session, the experimenter gave the user feedback on her performance for the assigned tasks giving her an idea about the extent to which she had completed them. This was done to provide the user a chance to rescale her efforts in finding the answers, if necessary.

The experiment session was timed with a limit of 1 hour. The user was alerted verbally by the experimenter at 30 minute and 45 minute mark running the experiment. The user was assigned 5 tasks based on a series of 4 related meeting conversations (ES2008 series of AMI corpus) on the design of new remote control. The 4 conversations were displayed on separate tabs, each tab containing all the basic controls and views as show in figure 3.4. The settings and controls on one tab worked independently of the controls and settings on other tabs i.e. the changes in selection on one tab did not get reflected on the other tabs for the other meeting conversations in the series. All four conversations had the same group of four people participating with very specific roles. The conversations ranged in length from a few hundred to more than a thousand sentences (363 sentences in the kickoff meeting ES2008a, 922 sentences in the conceptual design meeting ES2008b, 897 sentences in the detailed design meeting ES2008c, 1421 sentences in the evaluation meeting ES2008d). The complete list of tasks assigned to the users can be found in appendix A.3 but we are discussing a sample task below in section 4.2.3 to give an example of what type of tasks we set for the users and what were our expectations

as their solutions. The 5 tasks in the experiment session were designed so that they could be answered independently of each other. So, the users were free to answer them in any order they preferred and we instructed them accordingly. They were also made aware that some of the tasks might involve browsing through multiple conversations in the series of meetings to glean the information required to fully answer the questions.

At the end of the experiment we administered a questionnaire to get feedback on the usability of the interface and to get suggestions to improve the interface (refer to appendix A.4). The questionnaire consisted of Likert scale questions with scale value ranging from 1 to 5 as well as open ended questions. The experimenter also held a discussion with each participant about their feedback on the questionnaire to get clarifications, if required. In general, the participants used the entities listed in the Entity View and keywords typed into the generic Keyword Search Box as entry points into the conversation. For navigation within the transcript people relied more on vertically scrolling in the Transcript View and on using the link between the extractive summary in the Summary View and the Transcript View. The detailed results are presented in the next chapter.

All the sessions were carried out on a standard running Windows 7 machine with 6 GB RAM, a 23 inch monitor and standard keyboard and mouse devices.

4.2.3 A Sample Task

For the first task, we set up a scenario where the Marketing Division wanted to analyze the quarterly sales report to find out whether the remote control launched was in keeping with the forecast by the project team responsible for the design. We asked the participants to find out what the target cost of each remote control unit was and what the target final consumer price was. We also requested the participants to find out what was the total amount the company was targeting to earn from this product and which design team member mentioned the target amount first. We requested the participants to copy the original sentences from the meeting transcripts to the answer pad (a basic text editor provided on a fifth tab on the interface) in addition to writing down the answer in their own words. We designed our user study this way since different people have different ways of aggregating

information and phrasing the same facts; as such, judging the responses and interpreting what the participants tried to convey become subjective for a human judge. This issue was especially of concern to us since the majority of our study participants were non-native English speakers. We instructed the judges to inspect the copied transcript sentences in the case a user missed some vital information before incurring a penalty. The details on how the participants' responses were judged can be found in the section 4.2.4. The tasks on the user study were designed so that the participants had to look for very specific information. For this particular task, we were looking for the phrases 'target cost twelve point five euro', 'sales price twenty five euro', 'aimed profit fifteen million euro' and 'first mentioned by the project manager' and their variants. A user who missed out on any of these 4 facts was penalized 0.5 points for every missed piece of information. So, the task was equal to 2 points in total out of the 12 points for the entire set of 5 tasks.

4.2.4 Marking Scheme

We recruited two native English speaking graduate students as judges to evaluate the participants' answers. The marking scheme was decided at a meeting among the judges and the experimenter. Based on that marking scheme the judges worked independently to assign a score to each of the participant's response. The independent sets of scores from the two judges showed a Pearson's correlation coefficient of 0.93 and the average of the two scores were used in the final analysis to find correlation between usage of different components of the interface and the participant's score. The summary statistics for the entire set of tasks assigned to the participants at the experiment session can be found in table 4.2.

Task No.	Total Points	Avg. Points Achieved	Std. Dev.	Median	Percentage Achieved
Task 1	2	1.70	0.55	2.00	85
Task 2	3	2.17	0.81	2.50	72
Task 3	3	2.33	0.78	2.50	78
Task 4	2	1.45	0.70	1.75	72
Task 5	2	1.72	0.46	2.00	86

Table 4.2: Summary statistics for task wise marking for the user study

The participants in general performed reasonably well for the entire set of 5

tasks assigned; scoring in a range of 5.75 to 12 out of a total of 12 points.

4.2.5 Our Hypothesis

We hypothesized that a user, who relied on using the ontology concepts presented in the Entity View and the Ontology View, rather than reading through the entire transcript by scrolling in the Transcript View, would perform better. To test this hypothesis we logged the number of times a user selected an entity, a speaker or a DA type node. We also logged the number of times and the unit amount the user employed the vertical scroll in the Transcript View.

Chapter 5

Results

The goal of our user study (presented in section 4.2) was to assess the usability and usefulness of different components of the interface and to find out whether people who performed better at the assigned tasks exhibited any behavioural trends or whether there were any behavioural patterns for the people who performed worse than others. In sections 5.1 and 5.2 we discuss the summary statistics of the scores achieved by the participants. In section 5.3 we analyze whether the participants background (education, age etc.) had an effect on her performance and whether some of these aspects acted as confounding factors. In section 5.4 we analyze the participants' feedback on the usefulness of different components and highlight behavioural trends and their effect on the score achieved. We also discuss other interactional behaviour we observed by analyzing the automatic logs saved during the experiment session in section 5.5.

5.1 Summary of Participants' Scores

Table 5.1 shows the summary statistics for the answers submitted by our 30 participants during the user study as judged independently by our two judges. The two sets of scores assigned by the two judges have a Pearson's correlation coefficient of 0.93 indicating a near perfect positive correlation (see Fig 5.1). So, we decided to use the average of the two sets of scores for further analysis.

To find the frequency distribution (see Fig 5.2) of the average of the two scores

Evaluator	Min	Max	Mean	Median	Std Dev
Judge 1	1.500	12.000	9.083	10.000	2.883
Judge 2	1.500	12.000	8.650	9.000	2.886
Final (Avg. of Judge 1 and 2 scores)	1.750	12.000	8.867	9.500	2.834

Table 5.1: Summary statistics for scores of the 30 participants

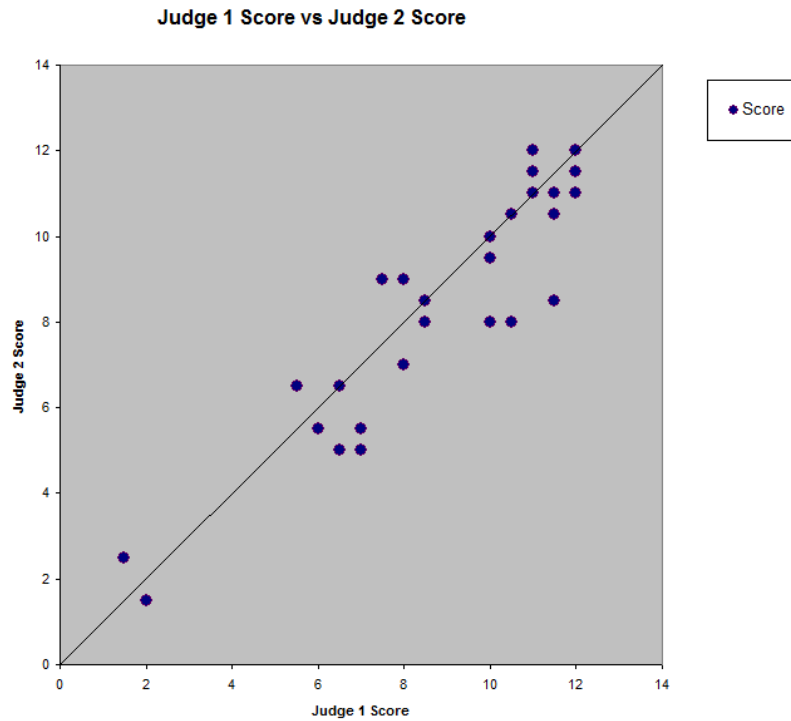


Figure 5.1: Score assigned by judge 1 vs score assigned by judge 2

with precision of two decimal places, we rounded each participant's final score (average of the two scores assigned by the judges) to the nearest integer score and then binned them. The resultant distribution shown in Figure 5.2 is a negatively skewed distribution with mode at score 11 and at score 12. Based on the summary statistics from Table 5.1, we decided to exclude the two participants whose scores

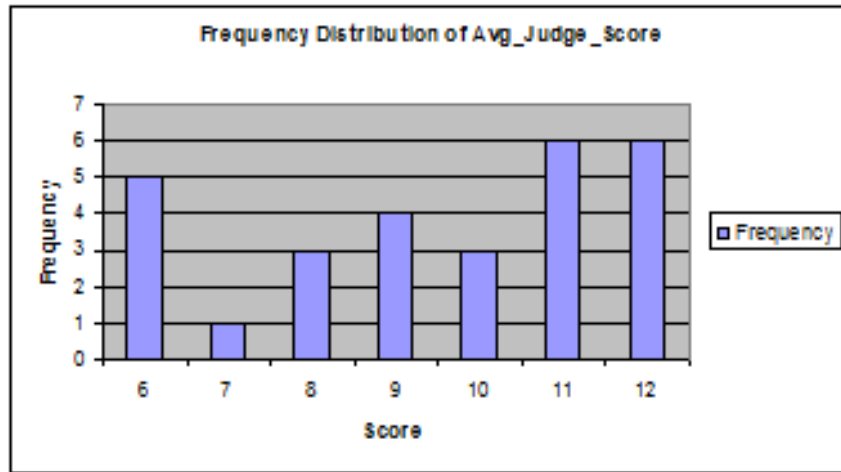


Figure 5.2: User study score frequency distribution

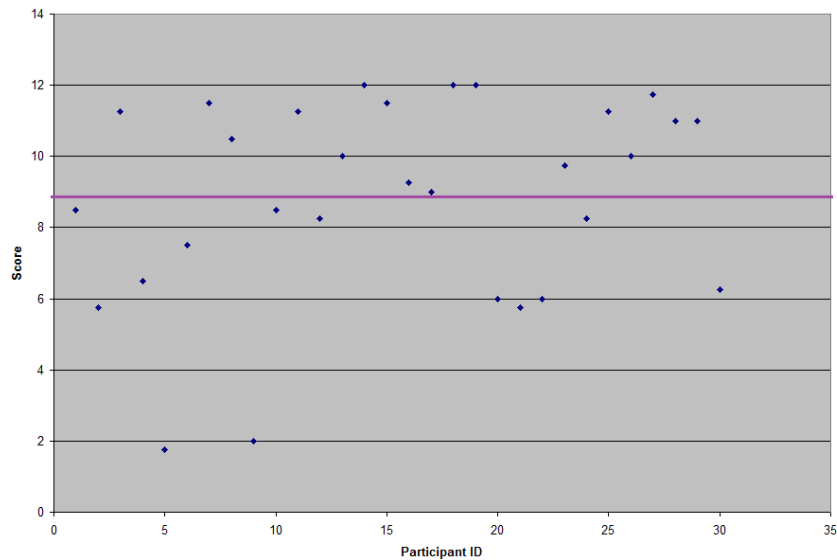


Figure 5.3: Average score of the thirty participants shown using a line, points represent individual scores

fell below the mean by more than two standard deviations as outliers (see Fig 5.3). We hypothesize that they performed well below the average since they were both non-native English speakers, they had been in an English speaking environment

Evaluator	Min	Max	Mean	Median	Std Dev
Judge 1	5.500	12.000	9.607	10.250	2.157
Judge 2	5.000	12.000	9.125	9.250	2.328
Final (Avg. of Judge 1 and 2 scores)	5.750	12.000	9.366	9.875	2.179

Table 5.2: Summary statistics for score of the 28 participants (excluding 2 outliers)

for less than six months and were not actively pursuing study or a job at the time of the user study. The other participants recruited for the study were either native speakers or were non-native speaking students who had to use English on a regular basis.

5.2 Revised Summary of Participants' Scores

Table 5.2 shows the summary statistics for the scores of the 28 participants we retained after removing the 2 outliers. This set of scores show a Pearson's correlation of 0.89 which is still considerably strong positive correlation.

5.2.1 Grouping of Participants Based on Scores Achieved

We categorized our 28 participants into 3 mutually exclusive groups based on the scores they achieved. The scores of the entire set of participants ranged from 5.75 to 12. Both the high performer group and the low performer group had nine people each from the higher end and the lower end of the score range respectively. By not including the people who performed moderately in either group, we ensured there was a clear distinction between the groups of high performers and low performers, leaving ten people in the moderately performer group who scored in the mid range. While defining the high performer and the low performer groups we took into consideration several criteria. We tried to keep the groups large enough for statistical validity and made sure that the definitions of the groups were clear in terms of the scores i.e. people who achieved the same score fell into the same group. As a result, the people who had an average score greater than 11 (a score of 11.25, 11.5, 11.75, or 12) were included in the high performer group while the people who had an average score less than 8.5 (a score of 5.75, 6, 6.25, 6.5, 7.5, or

8.25) were included in the low performer group. The rest of the participants (who scored between 8.5 and 11 i.e. 8.5, 9, 9.25, 9.75, 10, 10.5 or 11) fell under the moderately performer group. We had three participants who achieved full marks from both judges and had a perfect score of 12.

5.3 Pre-questionnaire Results

Before conducting the experiment session, we administered a pre-questionnaire to gather some background information on the participants. The complete set of questions can be found in appendix A.2. We have already presented the summary for the gathered information in section 4.2.1. After excluding the two participants whose scores were outliers, to ensure that there were no confounding factors we performed a two-sample t-test for the group of high scorers and the group of low scorers (see section 5.2.1 for the definition of these groups) for each of the background information aspect that we inspected. The t-tests showed no statistically significant correlation between performance of the participants and their gender, education level, comfort level in English, computer proficiency or familiarity with Information Visualization of large scale data. Still, we are enumerating the results of the t-tests below. For all of the tests the alternative hypothesis was that the true difference in means is greater than zero for the high performer and the low performer groups and the significance level was taken as 0.05.

A two-sample t-test was conducted to find out whether there is a gender bias for the high performer and the low performer groups. Gender is a categorical attribute and we assigned the value 1 to category ‘male’ and 2 to ‘female’ to perform the test. There was no significant difference in the scores for high performer ($M=1.22$, $SD=0.44$, $MED=1$) and low performer ($M=1.44$, $SD=0.53$, $MED=1$) conditions; $t=-0.9701$, $df=15.517$, $p=0.8266$. These results suggest that the gender of the participant did not play a role on the score achieved.

We conducted a two-sample t-test to find out whether older participants performed better at the user study than younger ones. There was no significant difference in the scores for high performer ($M=25.44$, $SD=3.61$) and low performer ($M=25.33$, $SD=2.60$) conditions; $t=-0.0750$, $df=14.536$, $p = 0.5294$. These results suggest that the age of the participant does not have an effect on the score achieved

by him or her.

We conducted a two-sample t-test to determine whether having a higher education level resulted in a higher score. The participants reported having an education level ranging from going to some university to pursuing a Ph.D. degree (see appendix A.2 for a complete list of input values). The education level is an ordinal parameter and we equated values (a) some university=1, (b) Bachelors degree=2, (c) Masters degree=2 and (d) Ph.D.=4 to perform our test. There was no significant difference in the scores for high performer ($M=2.78$, $SD=1.09$, $MED=3$) and low performer ($M=2.89$, $SD=0.60$, $MED=3$) conditions; $t=-0.2673$, $df=12.432$, $p=0.6032$. These results indicate that the highest degree attained (or being pursued) by a participant does not have an effect on her achieved score.

A two-sample t-test was conducted to find out whether being more comfortable speaking English resulted in a higher score. The participants reported having an average to excellent command of English (see appendix A.2 for a complete list of possible values). This 'English proficiency' attribute is ordinal in nature and we assigned the following numerical values to the inputs to perform our t-test : (a) Average=1, (b) Good=2, and (c) Excellent=3. There was no significant difference in the scores for high performer ($M=2.11$, $SD=0.60$, $MED=2$) and low performer ($M=1.67$, $SD=0.50$, $MED=2$) conditions; $t=1.7056$, $df=15.488$, $p=0.0540$. Although these results are not statistically significant at significance level 0.05, they come very close. So, we conclude that high performers in general self-reported to be more proficient in English than the low performers. The conversations the participants analyzed during the user study were generic and did not involve many domain-specific terms. However, the tasks were designed to give the participants clues on what types of sentences (decision or problem etc.) to inspect closer, or what would be a possible list of keywords to search. Having a better command of English has probably helped the participants to easily pick up these clues.

A two-sample t-test was conducted to find out whether the self-reported comfort level with using computers (in a scale from 1 to 10 where 10 means highly comfortable using a computer and 1 means not comfortable at all) by the participants had an effect on the score. There was no significant difference in the scores for high performer ($M=8.44$, $SD=1.13$) and low performer ($M=8.11$, $SD=1.27$) conditions; $t=0.5883$, $df=15.79$, $p=0.2823$. These results suggest that the self-

reported comfort level using computer by the participants does not have an effect on their achieved score.

We conducted a two-sample t-test to examine whether people who spent more hours per day on a computer did better than others in the user study. There was no significant difference in the scores for high performer ($M=9.22$, $SD=3.87$) and low performer ($M=6.44$, $SD=3.57$) conditions; $t=1.5827$, $df=15.903$, $p=0.06658$. These results are not statistically significant at 0.05 significance level but come close. So, people who spend more time on computers on a daily basis did somewhat better than people who rarely spent time on computers everyday. This effect is expected since although our interface did not involve the use of any other software application, it is heavily dependent on the interaction between the participant and interface components using standard input devices, and spending considerable time on a computer everyday makes use of these devices easier and intuitive.

We conducted a two-sample t-test to determine whether playing video games regularly had an effect on the score since we hypothesized that people who played video games a lot would have a faster reaction time and better visual perception and thus would perform better. This attribute has categorical values of 'Yes' and 'No' and we assigned the numerical value of 2 for 'Yes' responses and 1 for 'No' responses. There was no significant difference in the scores for high performer ($M=1.11$, $SD=0.33$, $MED=1$) and low performer ($M=1.33$, $SD=0.50$, $MED=1$) conditions; $t=-1.1094$, $df=13.938$, $p=0.857$. These results indicate that playing interactive video games did not affect the participants' attained scores.

A two-sample t-test was conducted to find out whether the familiarity level (in a scale of 1 to 10) with large scale data visualization had an effect on the score. There was no significant difference in the scores for high performer ($M=3.44$, $SD=2.70$) and low performer ($M=3.11$, $SD=2.67$) conditions; $t=0.2636$, $df=15.998$, $p=0.3977$. These results indicate that the self-reported familiarity level with large scale data visualization had no effect on the participants' scores.

5.4 Post-questionnaire Results

After completing the experiment, we asked the participants of the user study to fill up a post-questionnaire to give us feedback on the usability of the interface and to

gather suggestions on how to improve the interface. Refer to appendix A.4 for the complete questionnaire. The questionnaire included a number of Likert scale questions on the usability of the interface. We also asked a number of open questions to gather the participants' feedback on how useful they found the different components of the interface and whether they had any comments on how to modify the interface to help accomplish the tasks more efficiently.

5.4.1 Likert Scale Questions on Usability of the Interface

The questionnaire had 9 Likert scale questions with a scale ranging from 1 to 5. We altered the direction of the scale occasionally, sometimes associating scale value 1 with 'agree strongly' and 5 with 'disagree strongly' and sometimes the other way, to ensure that the participant did not fall into a trend of selecting a particular value for all the Likert scale questions. Figure 5.4 and Table 5.3 shows the summary for the 9 Likert scale questions we asked on the post-questionnaire. For ease of interpretation of the statistics, i.e. whether a value indicates positive impression among the participants or negative, we are restating the questions and the scale directions here.

1. I found the conversation browser intuitive and easy to use. (disagree strongly=1, agree strongly=5)
2. I was able to find all of the information I needed. (disagree strongly=1, agree strongly=5)
3. I was able to find the relevant information quickly and efficiently. (disagree strongly=1, agree strongly=5)
4. I feel that I completed the task in its entirety. (agree strongly=1, disagree strongly=5)
5. The task required a great deal of effort. (agree strongly=1, disagree strongly=5)
6. I felt I was working under pressure. (disagree strongly=1, agree strongly=5)
7. I had the necessary tools to complete the task efficiently. (agree strongly=1, disagree strongly=5)
8. I would have liked the conversation browser to have contained additional information about the conversations. (disagree strongly=1, agree strongly=5)
9. The interface quickly reflected the changes caused by interaction (changes caused when you select or unselect tags etc.) (disagree strongly=1, agree strongly=5)

The mean of the response for question 1 ($M=3.714$) shows a value that deviates significantly (more than 0.5 points) from the midpoint of the 5-point Likert scale.

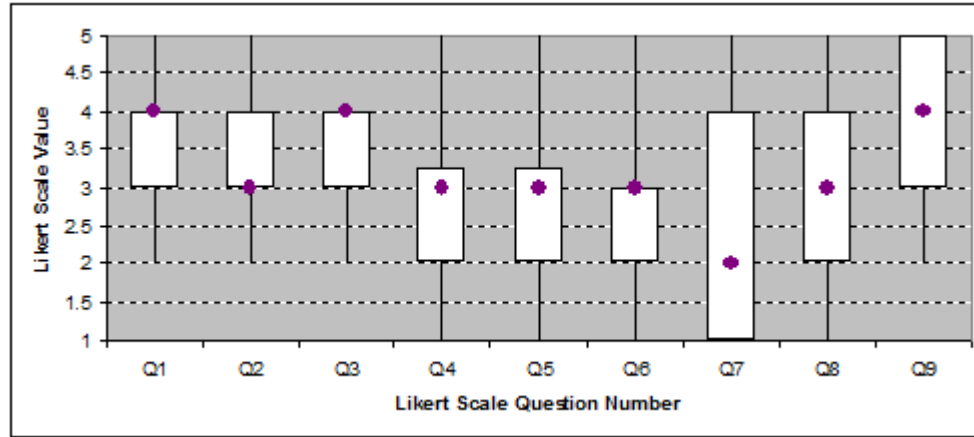


Figure 5.4: Box and Whisker plots for the questions 1 to 9

Question No.	Min	Max	Mean	Std. Dev.	Median
Q1	2	5	3.714	0.897	4
Q2	2	5	3.321	0.945	3
Q3	2	4	3.391	0.786	4
Q4	1	5	2.786	1.166	3
Q5	1	5	2.750	1.041	3
Q6	1	5	2.857	0.970	3
Q7	1	5	2.429	1.289	2
Q8	1	5	2.750	1.143	3
Q9	2	5	4.036	1.036	4

Table 5.3: Summary statistics for the post-questionnaire Likert scale questions

So, in general, users found the browser easy and intuitive to use. The median for this question is 4, also indicating an inclination of the users toward receiving the interface as intuitive. The participants also reported that they felt they were able to find the relevant information quickly and efficiently (median value 4 for question 3). They indicated that they had the necessary tools to complete the task efficiently (median value 2 and mean value 2.429 for question 7). The participants also reported the response time of the interface to reflect interaction changes to be good (median value 4 and mean value 4.036 for question 9).

We performed a series of two-sample t-tests for these Likert scale usability questions to investigate whether there were any statistically significant effects or trends seen among the group of high performers and the group of low performers (as defined in section 5.2.1). We are mentioning details of the tests that showed statistically significant results below.

We conducted a two-sample t-test to determine whether the high performers found the interface more intuitive and easy to use since we wanted to find out whether there was correlation between a participant's score and perception of the interface concepts. There was a significant difference in the scores for high performer ($M=4.11$, $SD=0.60$) and low performer ($M=3.33$, $SD=1.00$) conditions; $t=2$, $df=13.111$, $p=0.0333$. These results indicate that high performers felt that they had acquired a better understanding of the interface and grasped the concepts better than the low performers. To remove any variability in the instructions provided to different participants and to ensure that we explained the different components of our interface the same way to all of them, we had scripted the 3 sessions in the user study (tutorial, practice, and experiment sessions) using specific examples and followed the script as best as possible. Hence, we may claim that the difference in the perceived understanding of the interface by the two groups of participants has not occurred due to any external factor affecting their learning phase.

A two-sample t-test was conducted to find out whether the high performers felt that they were able to find all of the information they needed compared to the low performers. There was a statistically significant difference in the scores for high performer ($M=3.67$, $SD=1.00$) and low performer ($M=2.89$, $SD=0.78$) conditions; $t = 1.8383$, $df = 15.119$, $p\text{-value} = 0.04287$. These results indicate that the high performers had a higher satisfaction about the extent to which they were able to find useful information.

We conducted two-sample t-tests to examine whether the low performers felt like they were working under more pressure and whether the high performers felt like the interface provided necessary tools to complete the task efficiently. Although for these tests, there were no statistically significant results at significance level 0.05, the results came close. Both these t-tests show that the users had a good assessment of the entirety of the task assigned and their own level of accomplishment.

5.4.2 Feedback on Summary View and Its Link to the Transcript View

We asked the participants how useful they found the conversation summaries, both extractive and abstractive, and the linking of summary sentences to the transcript. Most participants mentioned the extractive summary to be very useful for narrowing down the search scope but did not find abstractive summaries to be helpful for browsing the conversation. They blamed lack of context for the abstractive summaries to be responsible for being unsuitable as a browsing tool. This is also apparent from the interaction data we automatically logged at the background of the interface. On average the participants inspected 34.93 extractive summary sentences further by clicking on it to auto-scroll the transcript to the corresponding sentence using their link and to peruse the context of the clicked sentence. There were only three participants who clicked on an extractive summary sentence less than 10 times and the person who used it most clicked on 123 extractive summary sentences during the experiment session. On the other hand, only four people tried out using the abstractive summary as means of navigation and they tested less than 10 sentences each. In the post-questionnaire three people explicitly mentioned that the abstractive summary is not well organized or useful, and needs more content. Six people said explicitly that conversation summaries (extractive) are useful tools for narrowing down search areas but in general the people mentioned it to be useful.

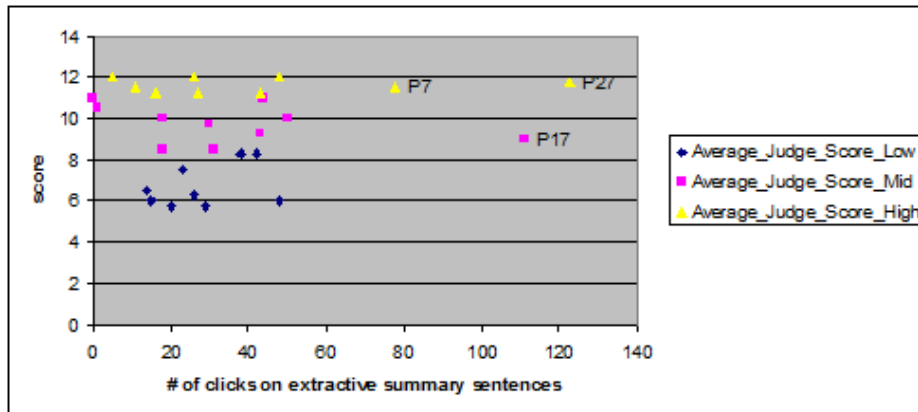


Figure 5.5: Scatter plot for the number of time participants clicked on the extractive summary sentences and their score

Out of the nine top scorers six found the extractive summaries very useful and the rest three found it moderately useful. Out of the ten mid range scorers two found the extractive summaries very useful, five found it moderately useful and the remaining three felt that the usefulness was hampered when too many sentences were displayed based on the tag selections. Out of the nine low performers six said the extractive summaries were very useful. One person mentioned that it was useful but she preferred using keyword search. One person said it was useful only for a small set of sentences. We also performed a two-sample t-test to find out whether the high performer group found the summaries more useful than the low performer group or vice versa. The results were not statistically significant. Refer to Fig. 5.6 for a plot summarizing these statistics.

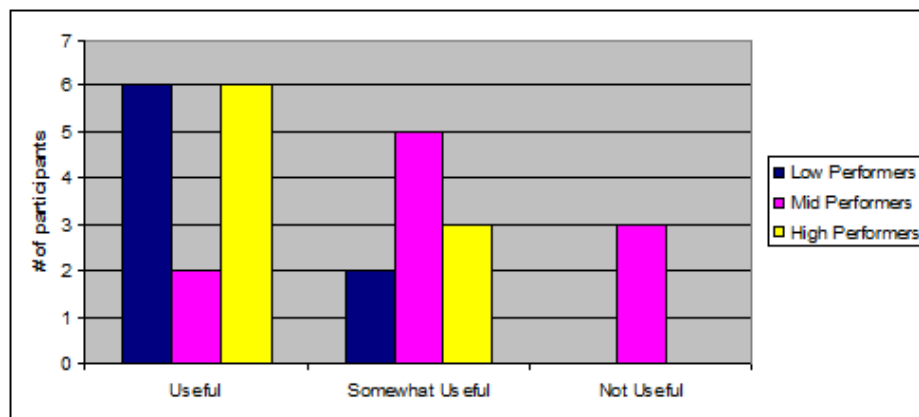


Figure 5.6: Column chart showing whether the participants from different performance groups found the extractive summary useful

Participants 7, 27 and 17 employed the link between extractive summary panel and the transcript heavily (see Fig. 5.5). Two of them (participants 7 and 27) were in the high scorer group and one of them (participant 17) performed moderately.

5.4.3 Feedback on the Ontology View

We asked the participants about the usefulness of the sentence speaker and sentence/DA type tags presented in the Ontology View in the post-questionnaire. There were mixed feelings about the filtering option provided by the concepts on the On-

tology View. Some participants found the Ontology View useful as query entry points, and some used it as a verification tool, some found it ineffective for the tasks set.

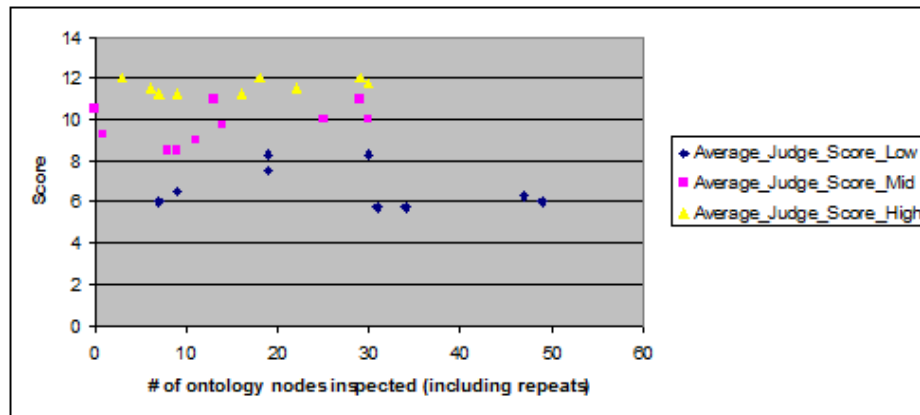


Figure 5.7: Scatter plot for the number of times a speaker or a DA type was selected in the Ontology View by a participant and the participant's score

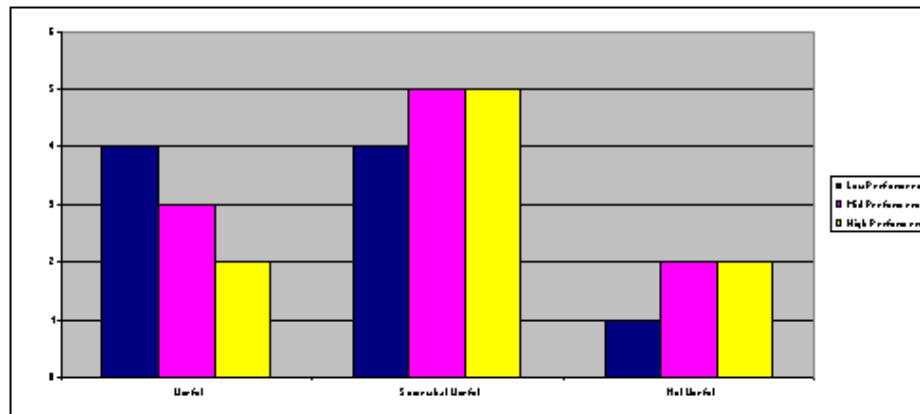


Figure 5.8: Column chart showing whether the participants from different performance groups found the Ontology View useful

In the high performer group two people didn't find the Ontology View useful at all, four found it moderately useful (one commenting on her strategy of using it as a second level of filtering after narrowing down the result set using entity tags),

two people found it useful, and one person found the DA type tag useful but not the speaker. In the mid-level performer group two people didn't find the component useful, four people said there was scope for improvement, three people stated that it was useful, one person found the DA type tag useful but not the speaker. In the low performer group one person didn't find the Ontology View useful, one found it moderately useful, four people found it useful, two people found the speaker tags useful but not the DA type tags, one person found the DA type tag useful but not the speaker (refer to Fig. 5.7). Refer to Fig. 5.8 for a plot summarizing these statistics.

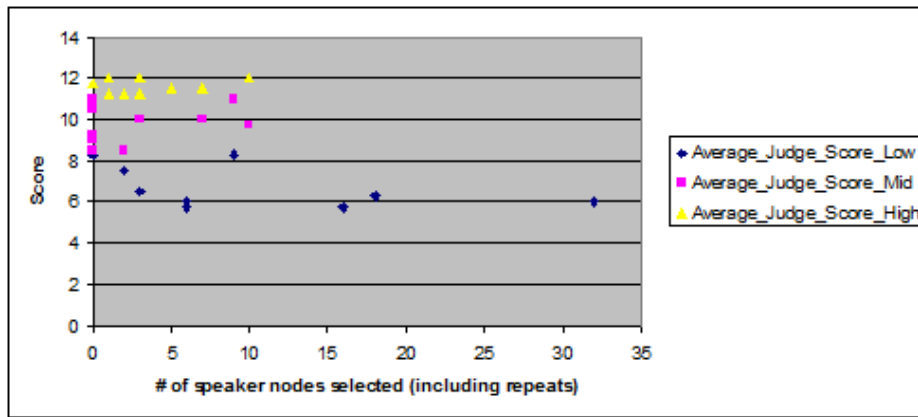


Figure 5.9: Scatter plot for the number of times a speaker is selected in the Ontology View by a participant and the participant's score

What we found interesting about the plots (see Fig. 5.9 and 5.10) for the number of times a speaker node was selected (including repeated selection of the same speaker and without repeat selection of distinct speaker nodes) is that people who investigated a smaller number of speakers did well. We believe, people who used their real-world knowledge and the cues provided by the wording in the task instructions to make informative decisions on who might be talking about the relevant issues (since the speakers in AMI meetings have very specific roles) comparatively performed well.

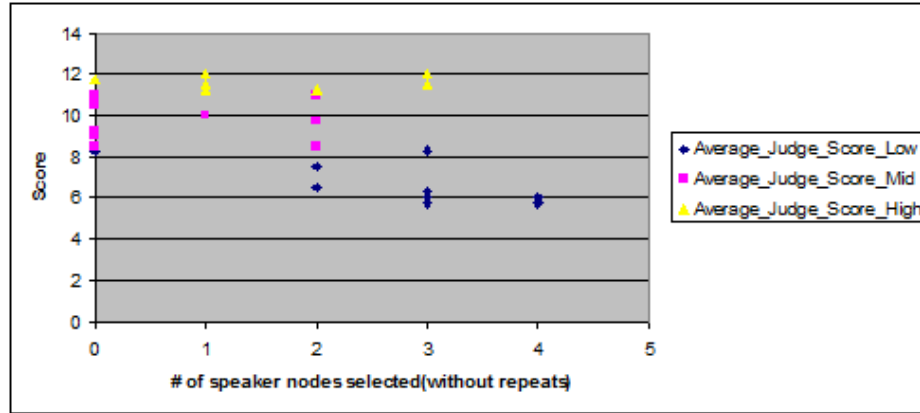


Figure 5.10: Scatter plot for the number of times a distinct speaker was selected in the Ontology View by a participant and the participant's score

5.4.4 Feedback on the Entity View

We asked the participants about the usefulness of list of topics in the Entity View. All but one of the high scorers mentioned the Entity View as a useful component, one person even dubbed it a 'life saver'. Among the people performing moderately seven people found the Entity View highly facilitating the tasks, two found it moderately useful. Among the people who scored in the low range, two people found the Entity View ineffective while four found it useful, and three found it fairly useful. Refer to Fig. 5.13 for a plot summarizing these statistics.

Participant 27 in the high scorer group, participants 23 and 26 in moderately performer group, and participants 2, 20, and 30 from the low performer group relied heavily on browsing the conversation by using entities as entry points (see Fig. 5.11). The plot shows the number of times a participant selected an entity (may deselect and select a particular entity again) vs. the score she received. Participant 27 in the high scorer group, participants 23 and 26 in moderately performer group, and participant 30 from the low performer group also selected the largest set of distinct entities while performing the tasks (see Fig. 5.12). The plot shows the number of times a participant selected an entity (may deselect and select a particular entity again) vs. the score she received. These two plots indicate that exploring a wider number of entry points does not necessarily result in a higher score.

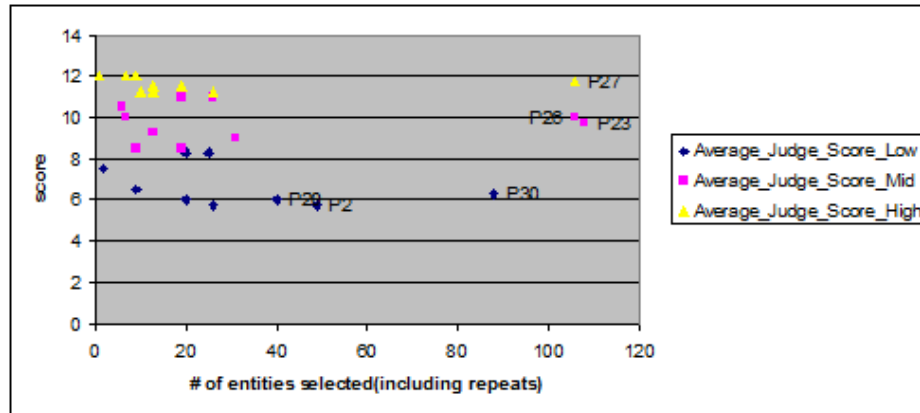


Figure 5.11: Scatter plot for the number of times an entity was selected by a participant and the participant's score

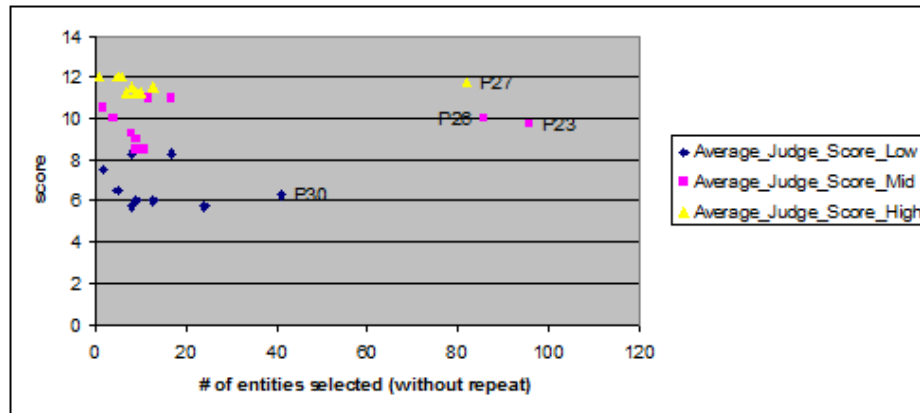


Figure 5.12: Scatter plot for the number of times a distinct entity was selected by a participant and the participant's score

Participant 27 who is one of our high performers also made use of the Entity Sort Order control to alternatively list the entities ordered by frequency and alphabetically (see Fig. 5.14). However, this participant did not make as much use of the range slider to shorten the list of entities (see Fig. 5.15).

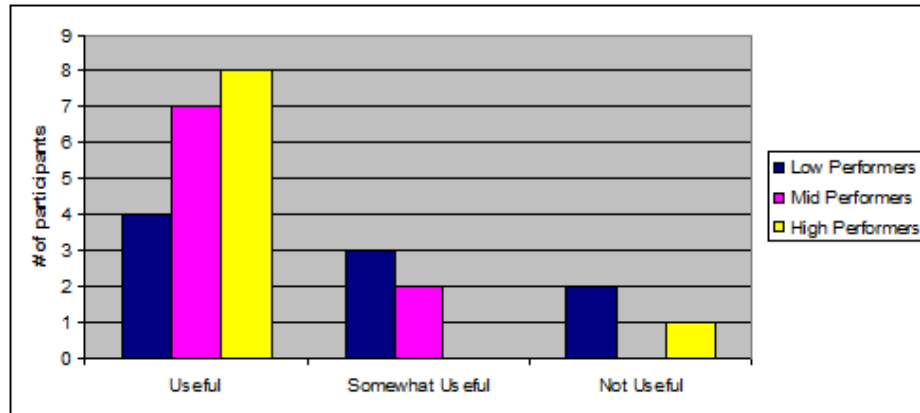


Figure 5.13: Column chart showing whether the participants from different performance groups found the Entity View useful

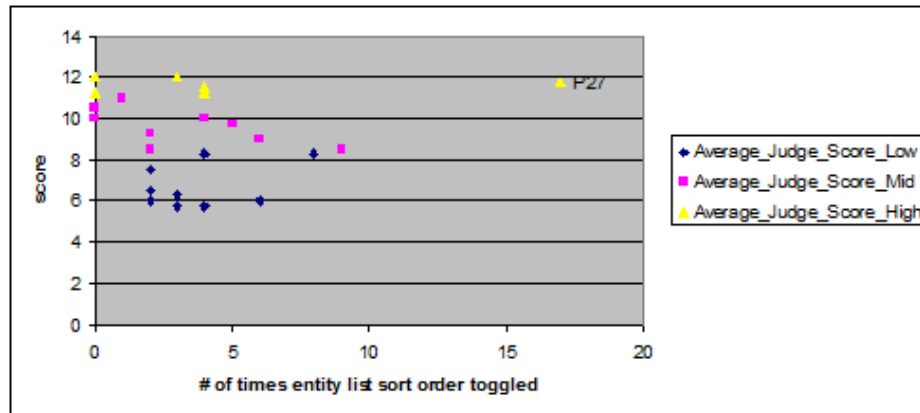


Figure 5.14: Scatter plot for the number of times sort order of the Entity View was toggled by a participant and the participant's score

5.4.5 Feedback on the Tag Selection Settings

In general, people who depended heavily on the concepts presented in the Entity View and the Ontology view also used this particular control frequently. The the number of time Tag Selection Settings was toggled has a Pearson's correlation coefficient of 0.641 (significant at 0.01 level) with the number of nodes selected on the Ontology View, a coefficient of 0.378 (significant at 0.05 level) with the number of nodes selected on the Entity View. For specifically DA type and speaker

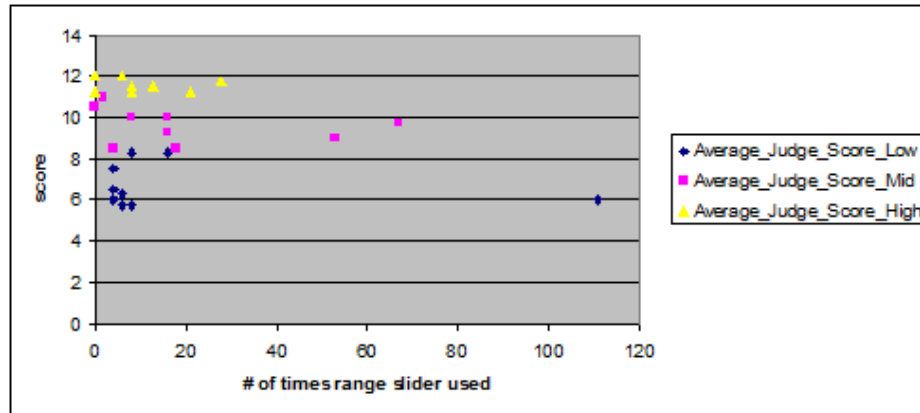


Figure 5.15: Scatter plot for the number of times range slider of the Entity View was used by a participant and the participant's score

nodes the correlation coefficients are 0.423 (significant at 0.05 level) and 0.652 (significant at 0.01 level) respectively. Five of our participants explicitly made comments on how Tag Selection Settings are useful for focused queries and have the ability to narrow down search using overlap of tags.

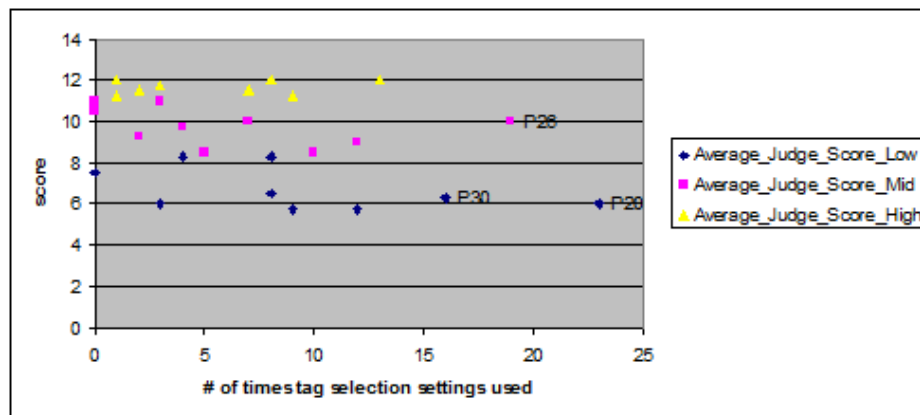


Figure 5.16: Scatter plot for the number of times tag selection settings was toggled and participants' score

Out of the nine top scorers one participant found the Tag Selection Settings very useful, five people found it moderately useful (one commented on more flex-

ibility in selection criteria i.e. user's control over which particular tags need to be considered) found it moderately useful, three people said it was not useful at all. Out of the ten mid range scorers five found this feature very useful, four found it to be moderately useful (one person also commented on a providing an option to select a subset rather than all of the selected tags) and the remaining person preferred using the Entity View and its relevant controls. Out of the nine low performers four said this feature was very useful, four said it was moderately useful, one person found only the 'All of the selected tags' setting useful to accomplish the tasks. Refer to Fig. 5.17 for a plot summarizing these statistics.

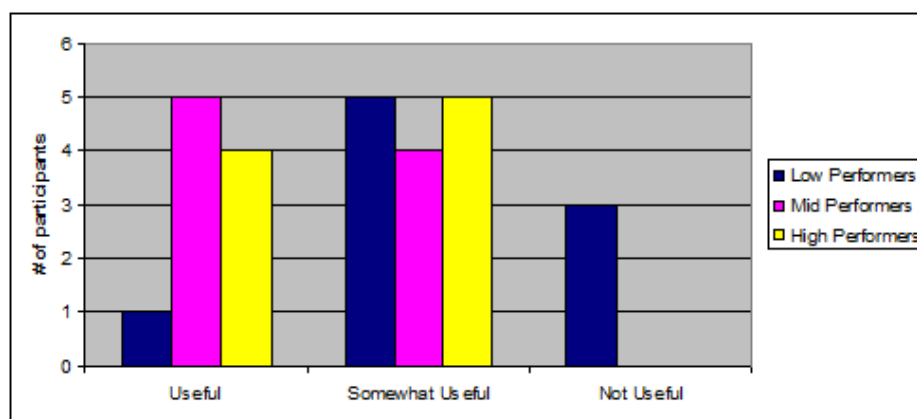


Figure 5.17: Column chart showing whether the participants from different performance groups found the Tag Selection Settings useful

Participants 26, 20 and 30 made use of the Tag Selection Settings more than others. Participant 26 performed moderately in the user study while participants 20 and 30 were in the low scorer group (refer to Fig. 5.16).

5.4.6 Feedback on the Marker Bars

Around half of the participants refrained from using the markers significantly (> 10 times) for navigation during the user study (refer to Fig. 5.18). We hypothesize that this happened because the marker bars are the most complex of the navigation tools provided on the interface and possibly users did not have sufficient time to learn how to effectively employ the distribution of markers along the bars to identify the

informative parts of the conversation. So, the marker bars involve a steep learning curve and we also need to address the issue of visual clutter for longer conversations. During the user study, three people suggested that the marker bars need titles, or a mouse-over description for the purpose of each one. Four people suggested to make the marker bars wider and to have more white space in between the bars. Four people remarked that the marker bars were useful for quickly searching through potentially useful sentences (navigation) and exploring surrounding sentences after locating a sentence of interest (browsing).

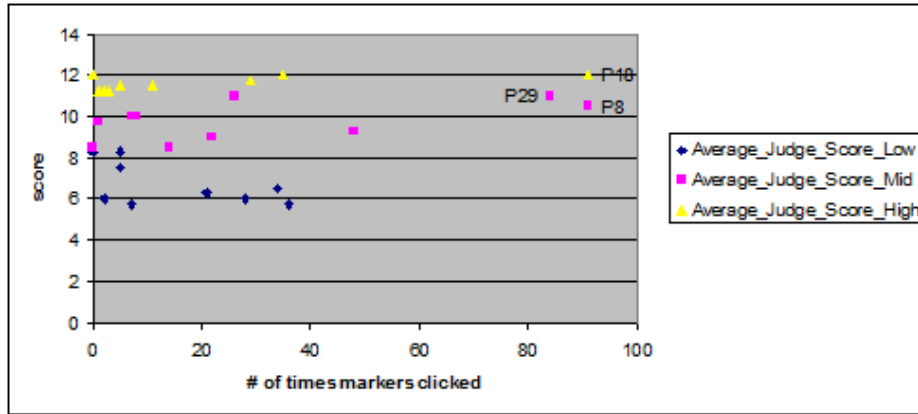


Figure 5.18: Scatter plot for the number of times markers clicked to navigate within conversation and participants' scores

In the high scorer group, three people found the marker bars unusable, two people found it very useful, three people found it too cluttered and commented on the usefulness of the tooltip texts. In the mid-level scorer group five people found the markers very useful, two people found it not useful at all and three people found the marker bar concept useful but unusable due to clutter in practice. In the low performer group, six people found this feature useful, two people did not like it and one person liked the tooltip texts. Refer to Fig. 5.19 for a plot summarizing these statistics.

Participant 18 (one of our three top performers who scored a full mark) and participants 8 and 29 (who scored in the mid-range) relied heavily on using markers as navigational tool.

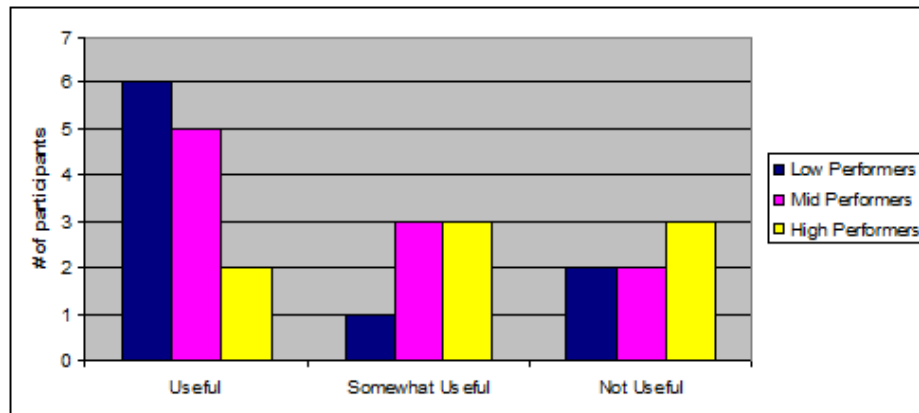


Figure 5.19: Column chart showing whether the participants from different performance groups found the Marker Bars useful

5.4.7 Feedback on Accuracy of the Entities Listed in Entity View

In our user study seven people commented that the Entity View misses out on important terms (low-frequency, critical words) that they were searching for while trying to complete the tasks. Other than these participants, in general, people found the listing complete and accurate. Four people remarked on the usefulness of the overview of conversation provided by the list in the Entity View and commented on using that list to find relevant information across multiple conversations.

Out of the nine people in the high performer group six found the listings to be very accurate, while three people found it somewhat accurate, missing out on some important terms. Out of the ten people in the mid-range performer group, four found the listings close to perfect, one person abstained from commenting, one person found it accurate but not complete, three people thought it missed out on some important terms, one person didn't notice since she was not using this feature frequently. Out of the nine people in the low performer group five found the listings to be very accurate, while four people found it somewhat accurate. Refer to Fig. 5.20 for a plot summarizing these statistics.

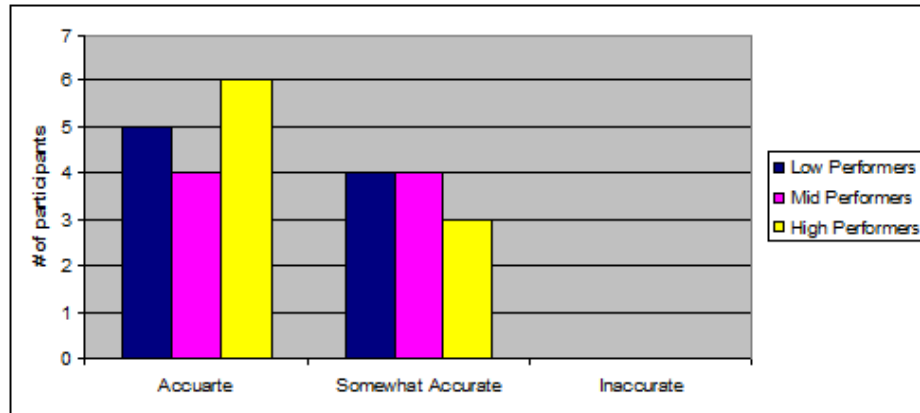


Figure 5.20: Column chart showing whether the participants from different performance groups found the listing on Entity View accurate

5.4.8 Feedback on Accuracy of the DA Type Tagging in the Ontology View

During the user study, there were three participants who asked for additional categories of DA types such as subcategorizing the ‘PositiveSubjective’ DA type to ‘Agreement’ and ‘PositiveOpinion’ etc. In general, people had difficulty grasping the difference between ‘ActionItem’ and ‘Decision’ sentences, between ‘Problem’ and ‘NegativeSubjective’ sentences etc. This difficulty had arisen for two main reasons. First of all, the definitions of the DA types are not easily comprehensible by general users. For example, ‘PositiveSubjective’ includes any implicit positive notion as well as positive opinions or comments. So, a sentence like “Okay.” expressing the speaker’s agreement with the previous statements is also classified as ‘PositiveSubjective’. The participants found these types of sentences disrupting since they unnecessarily made the set of tagged sentences shown in the Summary View large and did not provide any indication on what exactly was being talked about. Secondly, a sentence or an utterance can have multiple DA type tagging since different parts of the sentence can convey different senses. For example, the sentence “Let’s go with a simple chip.” would be classified as both a ‘Decision’ and a ‘PositiveSubjective’ because of the phrases “let’s” and “simple” respectively. Improving the accuracy of the DA type tagging, taking into consideration the hu-

man point of view of interpretation, is a major area that needs improvement in our future work.

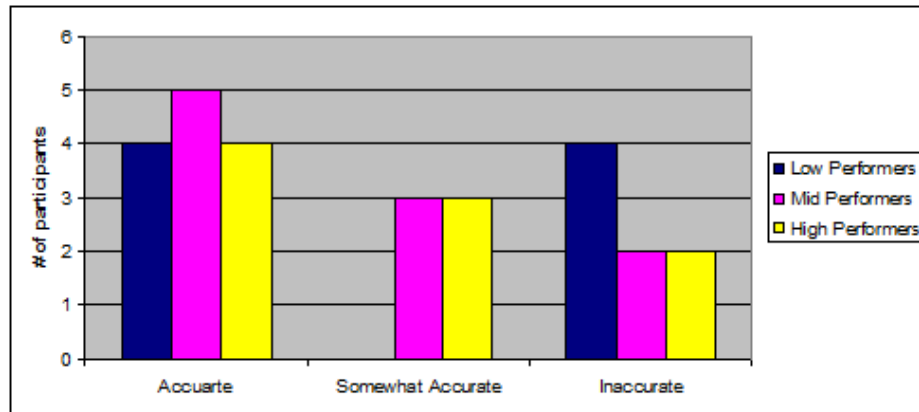


Figure 5.21: Column chart showing whether the participants from different performance groups found the DA type tagging accurate

Out of the nine people in the high performer group, four found the DA type listings close to perfect, three people found it moderately useful and two found it confusing. Out of the ten people in the moderately performing group, five people found the listings fairly accurate, one person said it was somewhat accurate, two felt the definitions were too broad, two people found the tagging confusing. Out of the nine people in the low performer group, four found the listings quite accurate, one person didn't find it useful so abstained from commenting, and four people commented on the lack of accuracy in tagging especially for 'PositiveSubjective', 'Decision' and 'Problem' type sentences. Refer to figure 5.21 for summary of these statistics.

5.4.9 General Comments

We had also requested the participants to provide us suggestions on how to improve the interface. Three of the participants wanted to see the speaker tags appear in different colors in the Transcript View. This solution was considered during the second redesign phase (see section 3.3) but later discarded due to scalability issues. We have discussed a possible approach to address this issue in the future in section

6.2.1. 5 people mentioned the need for an overview window which shows the ontology concepts as well as searched keyword across all conversations in a thread in a single window (for the 4 related meetings displayed on separate tabs during the experiment, refer to section 4.2.2). Refer to section ?? for details of our proposed extension based on this requirement.

5.5 Other Interactional Behaviour

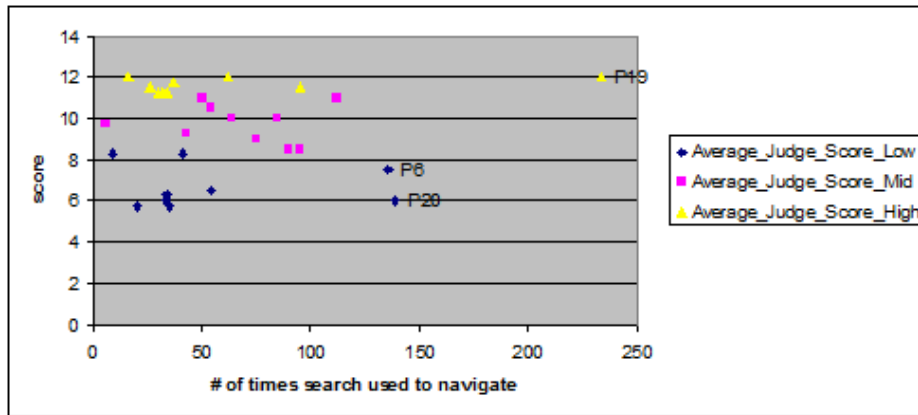


Figure 5.22: Scatter plot for the number of time search button used to navigate and participant's score

From figure 5.22 it is apparent that participant 19, who is one of our perfect scorers used the Search button to navigate within the transcript more frequently than other users. However, the two other participants who stand out for similar usage pattern are participants 6 and 20 who fall under the low performer group.

In figure 5.23, participant 19 once again stands out for looking up the highest number of keywords using our generic Keyword Search Box feature. Participant 20 of the low performer group also looked up a large number of search terms as well as participants 8, 17, 28, and 29 in the mid-range scorer group. All these participants had looked up at least 20 distinct search terms throughout the experiment session.

In contrast, participants 15, 18, and 25 of the high performer group relied on vertically scrolling the Transcript View to find relevant information. The participants 10 and 29 from the mid-level performer group as well as participants 6 and

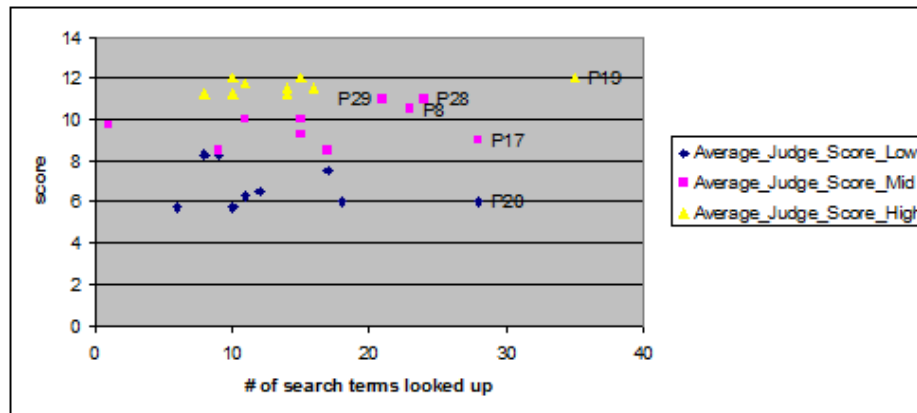


Figure 5.23: Scatter plot for the number of search terms looked up and participant's score

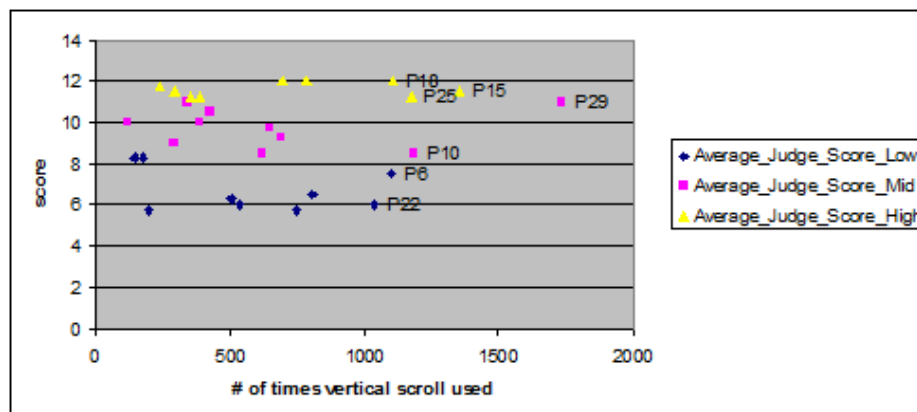


Figure 5.24: Scatter plot for the number of unit vertical scroll used in the Transcript View and participant's score

22 of the low performer group also stand out in figure 5.24 for their reliance on vertical scrolling for gathering information.

We had three participants who had received a perfect score at the user study - participants 14, 18, and 19. Participant 19 relied heavily on the generic Keyword Search Box (refer to figures 5.23 and 5.22) both as a navigation tool and to find entry point to the conversation. Participant 18, on the other hand, relied on marker bars for navigation within the transcript (refer to figure 5.18). Participant 27 who

also attained a near perfect score relied on the Entity View for filtering entry points to the conversation (refer to figure 5.11) and relied on the link between the extractive summary and the transcript for navigation (refer to figure 5.5). Participant 7 in the high performer group also relied heavily on the link between the extractive summary and the transcript for navigation (refer to figure 5.5). Participant 14, who is our third perfect scorer, never showed up as an outlier in any of these plots indicating that she did not rely explicitly on any of the components but rather made moderate use of different features.

Our discussion in this chapter indicates that there is no single best approach for browsing and summarizing a conversation using our interface. Rather, the choice has to be made by the user which features she finds more convenient and effective for the task on hand.

Chapter 6

Conclusion and Future Work

In this thesis, we presented a visual interface for interactively browsing multi-modal conversations and for automatically generating focused summaries for them. Our current interface is capable of handling linear, single session conversations. The next major extension to our interface would be adding the capability to analyze multi-session and non-linear conversations. During our user study (refer to section 4.2), we had presented the four related meeting conversations on separate tabs with completely separate controls. Having an overview window to facilitate searching or analyzing multiple related conversations simultaneously could potentially be a very useful addition to the interface. We could show a set of four marker bars for each conversation under consideration in this overview window and provide flexibility to move the marker bars around and to place the bars dedicated to a specific concept type for all the conversations side by side. This would be extremely helpful in comparative analysis. In the user study, the participants also mentioned the need for forwarding some control settings to all related conversations; for example, the users would have liked the search terms to be applied to all four meeting conversations when they type it in for any one of them. These feedbacks from the user study will be instrumental in the future redesign of our interface.

We have presented our findings from the user study that we have conducted to evaluate the interface in the previous chapter. Based on these results we now propose several extensions and modifications to our interface for the future in section 6.2.

6.1 Conclusions

In this thesis, we have presented two major redesigns over the proof-of-concept prototype presented in [10]. Our latest interface is a major improvement over the initial one in terms of use of HCI, NLP, and INFOVIS principles. From section 5.4, it is clear that different people take different approaches in navigating through a conversation and in finding informative sentences - some rely on the summary sentences, some on basic keyword search, some on the markers etc. These results show the necessity for including all these components even though there is a huge overlap among their functionality. Our interface caters to the widely varying approaches adopted by people to find informative parts of a conversation and can be used to analyze conversations of different modes and domains.

6.2 Future Work

Our current design has evolved through numerous iterations of small modifications applying HCI, NLP and INFOVIS principles. Still, there is much scope for further improvement. We present here a number of redesign ideas for our interface based on the feedback provided by the participants during the user study. We have also included a number of design rationales for making the interface more visually representative of the data and able to handle generic non-linear conversation threads (e.g., email conversations).

6.2.1 Color to Distinguish Speakers and Turns

In our design, the participant icons for the conversation are displayed all the time in the Speaker column of the Transcript View to keep the users oriented to which participant is contributing to the conversation at the moment. We use larger grid boxes to encompass the sentences in a turn by a specific participant in order to show their grouping using containment. Additionally, using a slight change in the background color for alternating turns might make it easier for the users to navigate. Another feedback from the participants was to use different colors for the icons of different speakers - a possibility that we had considered during the design phase but discarded due to scalability issues. Although for the AMI meeting corpus exactly four participants are involved in a particular conversation, in real life

scenarios, the number of speakers would vary widely. A user can barely distinguish more than a dozen colors at a time; and on top of using the color channel for the markers and for the DA type icons, we believe that the use of color to encode speakers as well would have been more problematic rather than helpful to the users. We could provide the user option to manually turn on/off coloring for the Speaker icons for a conversation with a scalable number of participants. An alternative could be to use a heavier line to draw the boundary of the grid boxes corresponding to a turn or to use different stipple patterns for the lines within a turn and for the lines separating alternating turns.

6.2.2 Family of Color Shading Scheme for Marker Bars

Some participants had difficulty keeping track of different selections in our interface using the two shade scheme for the Marker Bars where we showed the most recent selection in a darker shade and the rest in a lighter shade. A possible solution is to use a family of colors along a marker bar as suggested in the future work of ThemeRiver [21]. Since shades of a color would lose discernibility if too many of them are used, there will still be limitations to this solution; but it will be an improvement over current design. Another challenge is to maintain the shade of color for a particular concept through out the session. This will particularly be a problem for Entity type markers due to the large number of entities present in a normal length conversation. Although participants inspect a relatively small number of entities at a time to find the answer to a query, there are numerous combinations of the entities that they can try out. Also, during the study, users commented on putting more space in between the parallel bars and to make the bars wider to avoid accidentally clicking on a marker on an adjacent bar. A header over the bars indicating which concept type it represents would also be helpful for novice users.

6.2.3 Overview+Detail Approach for Marker Bar

Even after dedicating a marker bar for each concept type, for a long conversation there is still considerable visual clutter after selecting a small number of concepts of a particular type. In this case, magnification of a particular range of marker bar would help the user to interact with the markers easily and to see the distribution

pattern more clearly. We could have a rectangular lens like mechanism that could be slid along the length of the marker bars, and the set of markers falling under the scope would appear magnified in a separate set of marker bars, giving more pixels per sentence due to the reduced scope under consideration. This is similar to the idea of mapping the entire document to the right side of the window frame and the currently viewable portion of the document to the left side presented in [8].

6.2.4 Highlight Search Keywords within Transcript

Another suggestion was to highlight the matches for the search keyword within the transcript in a more prominent way using bright colors. The users also asked for the option to retain markers or results of previous search terms and the ability to search for sentences that refer to multiple search words (irrespective of the order of appearance in the sentence). Some of them suggested the option to treat the search keywords as normal tags and include the relevant sentences while using the Tag Selection Settings and the option to search only within a highlighted part of the conversation.

6.2.5 Filter Summary Path and Flexible Query Setting

The Tag Selection Settings along with different types of concept selection from the Ontology View and the Entity View can be used to find a good sized result set for further inspection. The user can toggle between the options ‘At least one of the selected tags’ and ‘All of the selected tags’ to get the union or intersection of the set of sentences tagged with a particular concept respectively. In the future, we could provide more flexibility here by letting the user override the set of concepts used for the Tag Selection Setting. We could borrow the idea of summary path from the Flamenco [55] and Mambo [16] systems to simplify the user interaction with the ontology selection set. In Flamenco, different paths may lead to a collection of images at a particular time; so Flamenco uses a summary path along the top of the interface to show exactly which path was taken and uses links along this path to retract to a previous decision along the path. Similarly, the Mambo system provides breadcrumb style filter history, which gives an interactive overview of the active facet filter. In our interface, to facilitate the inspection of a possibly

large ontology, nodes can be minimized (i.e., their children can be hidden) or the viewport may not be able to display the entire list for the ontology or entity list. So, it may happen that the set of tags selected by the users is not fully visible. Including a summary of the concept selection at the top of our interface, as it is done in Flamenco and Mambo, could be useful in providing an overview and could be used as a control for overriding tag selection criteria for complex query.

6.2.6 Information Scent for the Abstractive Summary

The abstractive summary for the sentences selected interactively using our interface is generated based on a NLG component. Presently, the ontology concepts can be inferred from the sentences generated but extracting them would be computationally expensive and redundant since they are already being done once in the NLG component. For example, the user can readily infer from an abstractive summary sentence like “The `UserInterfaceExpert` had negative opinions about the colours.” that the speaker of the component sentences that were aggregated to generate this summary sentence is the `UserInterfaceExpert`, the entity involved is ‘colours’ and the DA type of the component sentences is ‘NegativeSubjective’. However, to show icons corresponding to these concepts, we need to extract this information from the summary sentence or receive it as additional input from the NLG component. In the future, we shall modify the summary generation application to output the DA type, speaker and entity annotations as well as the list of transcript sentences used to generate a summary sentence. This will enable us to include information scents for the abstractive summary similar to the ones for extractive summary on current interface (refer to Fig. 3.5).

6.2.7 Dynamically Adjusting Entity List

For a normal sized conversation the list of entities in the Entity View is quite long ($\cong 100$) and cannot be handled easily by users. This sometimes makes the user resort to using the Search Box solely. A possible solution is to dynamically adjust the list of entities based on the keyword prefix the user types in the Search Box. For example, if the user starts typing ‘ba’ all the entities starting with it, like ‘battery’, or that include that expression, like ‘rechargeable battery’, would remain and the

rest would fade out. This would help the user to quickly find out whether the topic appears in the shortened Entity View. If it does, it is more convenient for the user to select that entity by clicking rather than having to type in the whole phrase in the Search Box. An alternate way to achieve the same effect would be to suggest probable entities to the user in a dynamically adjusting popup list when she starts typing a keyword in the Search Box like searching a keyword on Google [1].

6.2.8 Better Dynamic Layout

The screen real estate could be put to better use by using a more dynamic layout. In that case, some of the views and controls could be kept collapsed while the user is focusing on conversation analysis using particular widgets. For example, if a user wishes not to use the marker bars as her primary means of navigation or analysis, she could keep it completely minimized to allocate more space to the controls she finds more convenient to use. Moreover, the space allocated to the Ontology View could adjust automatically based on the number of concepts presented under the Speaker and the DA Type core nodes.

6.2.9 Repetition Pattern of Conversations and Clustering of Entities

Recurring patterns or phrases are common among individual posts in an email or blog thread or among the series of meetings in the AMI corpus. An overview of this type of repetitions may be helpful to the users to satisfy complex query needs. A way to analyze such recurring patterns has been discussed in [7].

There is a number of ways of grouping the entities that appear along the length of a conversation according to collocation, word meaning or high level topics of the conversation. The concepts presented in PhraseNet and WordTree could be extended to visualize such groupings of entities referred to in the conversation. Although the clustering approach for concept identification may or may not produce the term types and granularities useful to the user, it may be extremely beneficial in reducing cognitive load. So, as future work we are planning to investigate the presentation of the entities in our interface in logical groups. We discuss some possible approaches to clustering entities here.

[29] presents a semi-supervised algorithm that uses a root concept, a basic level

concept and recursive surface patterns to learn hyponym-hypernym pairs automatically from the web, position the concept based on web query and a graph-based algorithm that derives from scratch the integrated taxonomy structure of all the terms. Pattern based approaches are highly accurate but require a set of seeds and well-defined surface patterns. It also requires a large corpus like the web. Different tasks and criteria produce different taxonomies, even when using the same base level concepts. Attempts at producing a single multi-perspective taxonomy fail due to complexity of interaction among the perspectives. The major problem in performing taxonomy construction from scratch is that overall concept positioning is not trivial. This algorithm uses doubly-anchored lexico-syntactic patterns and bootstrapping to harvest terms and thus can adapt easily to different domains requiring minimum supervision. Once the terms are harvested a hypernym relationship direction between pairs of related terms is established by using a set of surface patterns and from the web hit counts the direction of the relationship is established.

[54] incorporates techniques from text mining, information retrieval, natural language processing and machine learning to generate a concept ontology. Nominal N-gram mining is used to identify the concepts. WordNet and surface text pattern matching are used to identify the relationships among the concepts. A supervised clustering algorithm is then used to further cluster the concepts based on pseudo-relevance feedback. Nominal N-gram mining consists of sentence segmentation, POS tagging, identifying sequence of nouns, mainly bigrams and trigrams and proper nouns where each word starts with a capitalized letter. The noun phrases with highest frequencies in the document are chosen as candidate noun phrases and their validity verified using a web query and threshold of hit count. The bigram concept candidates are organized into groups based on the first sense of the head noun in the WordNet. Trigrams are then compared with bigrams already in the hierarchy. If a bigram concept matches with the suffix of a trigram or named entity, the trigram or named entity is added as a child to that bigram. Webquery is then used to identify hypernym-hyponym relations among sibling concepts. The pairwise similarity score can be a linear function of some underlying features such as similarity of Web definitions of the two instances, similarity of sub-concepts, similarity of verb usage of the two instances etc. K-medoid clustering with sampling

is used to get the final hierarchy tree. A web-based approach is used for labelling the intermediate node concepts found in clustering.

6.2.10 Display Non-linear Conversations

In the future, we shall extend the interface to display non-linear asynchronous conversation threads. This modification entails including a preview of the entire thread and changing the Transcript View to accommodate multiple posts in the thread. We shall link each node in the preview to the Transcript View for that particular node and show detailed information like author of the text, timestamp, subject etc. of every comment or post when hovering a node in the preview [38]. In case of blogs, the interface could also provide a link to every comment in the forums original web interface. For the representation of the thread-structured transcript, we shall add a title bar or separator for each post. The title bar may contain different information that serves multiple purposes. Including the date of the post in the title bar could serve as a date separator like email archives where the emails are grouped according to the date received under a date title. We could use gradient coloring to indicate the amount of day shift between consecutive posts, a large shift in the gradients between the date separator of two consecutive posts would indicate a longer time gap between them [28]. We could also use color coding of the text to indicate topic drift in the post content. Since we are using color to highlight entities in the transcript, we could color code the separator instead of the text for this purpose to provide a high-level topic association. This could give users an indication of where, within a thread, the topic has drifted to an entirely different subject despite the persistence of the earlier topics subject line. The last post by the original author usually holds cues to what the conclusion is for the entire conversation thread. So, using color to highlight the last post may prove to be useful [43]. Even within a single thread the people involved may change, especially the recipient list of an email thread. We could mention the participant list for each post separately in the title bar for that post [4, 45]. Since displaying non-linear conversation threads will be a major extension to our interface in the future, we are discussing several options that we have explored to date in details below.

Asynchronous human conversations (blogs, emails etc.) often take a non-linear

form where the conversation thread forks out at some point of time. Due to limited display dimensions, vertical scrolling must be provided for long conversations, even when the conversation is fully linear. The branching of asynchronous conversation makes it harder to understand the overall content without knowledge of the thread structure. So, most of the times for asynchronous conversation visualization an Overview+Detail approach is followed, as in [45], ensuring that even if only a part of the conversation is visible in the details window, the user can refer to the overview pane to get the overall thread structure information and the position of currently viewable portion of the text within the entire thread. A thread overview also enables a user to get an idea about the global structure of the thread and to jump easily between branches [20]. It should be kept small because it has to be displayed in a space convenient way in its entirety [28], not interfering with the main features of the interface and interaction mechanism. There are different options for displaying the preview including thread view, arc view, dynamic thread view and node-plus tree structure as discussed below. Different approaches can be taken to indicate the transition between posts in a thread when displaying the transcript; we shall discuss a number of them later on in this section.

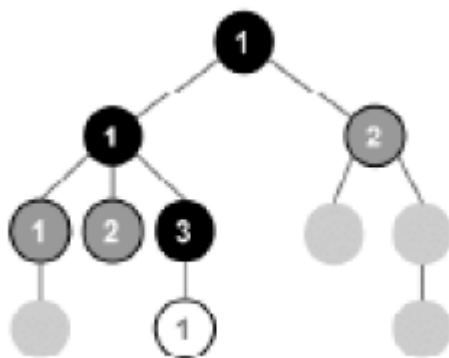


Figure 6.1: Thread View of a conversation thread with 11 posts in 4 levels; the numbering in the diagram traces the route from the root to the node in white labelled 1; the numbering restarts for each level of sibling nodes

In thread view, each post is represented as a node and the reply-to relationships are represented as edges between the parent and child post. This takes on a tree

structure with the original post as the root of the tree. The posts that are direct reply to the original post are presented as nodes in level one, the posts that are replies to nodes in level one are presented as nodes in level two and so on (see Fig. 6.1). This is the most common form of thread preview [20, 31, 38, 43] because it makes easy to detect out-degree of each node and the depth of each sub-tree possibly giving an indication of how controversial a topic was. However, this is a static view, the user cannot just concentrate on a sub-thread for a larger (deep or bushy) conversation thread; thus such an overview may run into space limitation. When a post has multiple parent nodes through reply-to relation and quotation (like emails), the graph takes on a network form rather than a tree form; the thread view does not take this into account.

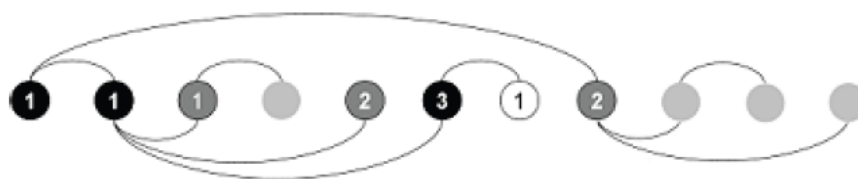


Figure 6.2: Arc View for the conversation thread presented in Fig. 6.1

In arc view [28], similar to thread view, each post is represented as a node and reply-to relationships are represented as edges between the parent and child post. However, the resultant graph is represented as an array of nodes linked by arc-like edges on two sides - the top and bottom of the nodes (see Fig. 6.2). When linking up the nodes, the objective is to keep arc overlap to a minimum. The height of the arc is thus proportional to the positional distance in the line up between the nodes being connected. Although the out-degree of a node is easily discernible from this view, the depth of a particular sub-tree is really hard to estimate at a glance. Instead, the user would have to start from the root node of the sub-tree and follow each arc which is a part of the sub-tree to traverse all alternate paths to compute the depth of the tree. Although due to the non-overlapping arc and linear arrangement of nodes, it may be easier to place the preview pane with an arc view in the interface, it is not possible to spatially group a number of nodes (using the tree level or sub-topic of conversation etc.).

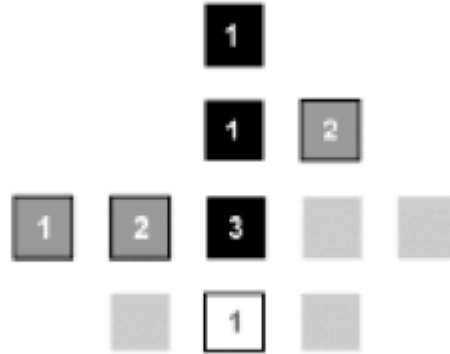


Figure 6.3: Dynamic Thread View for the conversation thread presented in Fig. 6.1

Instead of focusing on the entire thread structure or a particular sub-tree, the dynamic thread view in Conversation Map [20] focuses on the path from the root to the node of interest (see Fig. 6.3). The Conversation Map provides a generational structure overview of all the posts in a particular conversation. The postings in the thread are organized as rows of boxes, each box corresponding to a single post; with each row representing a generation of posts (based on depth of the post in the conversation thread) in the thread. The dynamic behaviour reveals more structural information about the thread as the user navigates from entry-to-entry. Each row is dynamically reconstructed as a posting is selected in the map. All postings in the branch of the selected posting (from selected node upto root) slide into the same column and are highlighted and surrounded by a vertical box.

The node-plus structure [20, 45] is in essence the windows folder explorer structure where each level of nodes is indented under its parent using a horizontal line and all siblings appear in the same vertical column in the order of their posting time and are associated to their parent using a vertical line originating from the parent (refer to Fig. 6.4). It is possible to keep the sub-tree rooted at any non-leaf node contracted to focus attention on the part of the thread that remains expanded. In the node-plus tree overview it is hard to get an estimate of node out-degree and tree depth at a glance. Nevertheless, this representation is more space efficient and

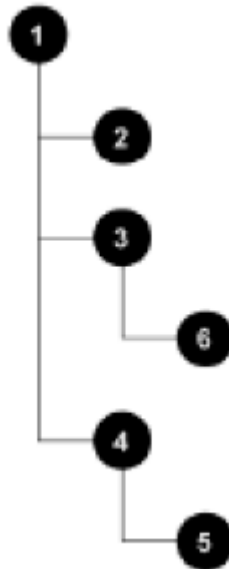


Figure 6.4: Node Plus structure showing parent-child relation using indentation and sibling relation using the same vertical column position

scalable for bushier conversations.

In radial trees [38], posts are represented as nodes and edges correspond to reply-to relations among them. The main post is used as the root of a tree and its direct replies are considered part of the first level and so on. The main discussion post that is chosen interactively is placed at the centre of the radial tree, while its comments surround it in a concentric manner, allowing the discovery of hot topics or user-to-user debates (refer to Fig. 6.5). Each level in the tree is depicted here using a circle to reinforce the notion of depth. There is no need to use pagination and nesting to navigate through multi-threaded conversations in this view; so, it's more efficient for highly discussed posts with several hundreds of comments. However, because of this concentric circle representation, it needs a considerable amount of space which makes it unsuitable for the overview pane for an application where the main objective is to focus on individual posts rather than the thread structure. This view can use different sizes, shapes, and colors to categorize nodes according to different metric values and provides details on demand regarding author of the

text, timestamp and body of every comment.

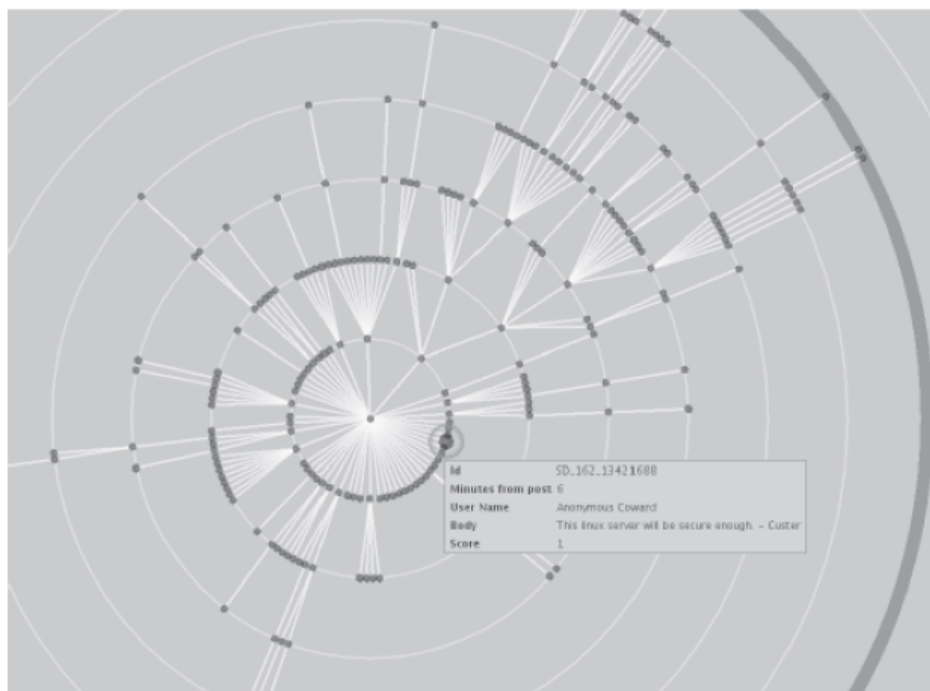


Figure 6.5: Radial Tree with root conversation session at centre and concentric circles indicating level of descent conversation sessions

In addition to showing an overall thread preview, it is also necessary to indicate transition from one post to the other in the Transcript View and to relate the sibling posts or ancestor-descendent posts. Same rules of indentation as the node-plus tree preview structure applies for the transcript indentation scheme [38]. Each post is shown as a separate box containing the text of the post and a title bar on the top of the box showing information about the post such as author, timestamp, rating etc (see Fig 6.6). A child post (or a reply) is indented by one level under the parent node and all the siblings are listed in order of the posting time from oldest to the newest. Without proper indentation (and even with proper indentation because of vertical scrolling) it becomes impossible to follow the thread structure in this view without referring to the preview pane and associating the posts with the nodes there.

Narrow Tree [36] is an incremental improvement to the linear indented trees fa-

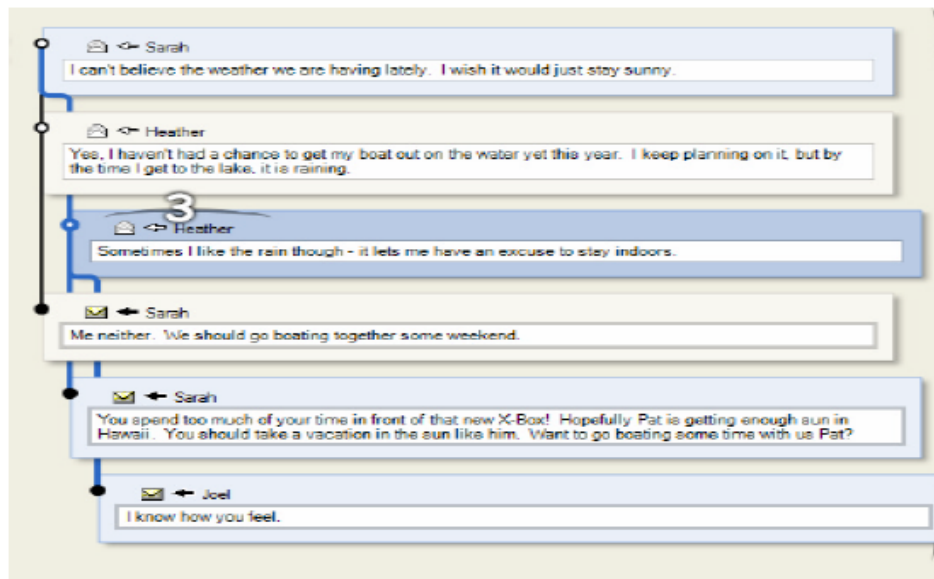


Figure 6.6: Node Plus structure for displaying transcript of the conversation sessions in the thread

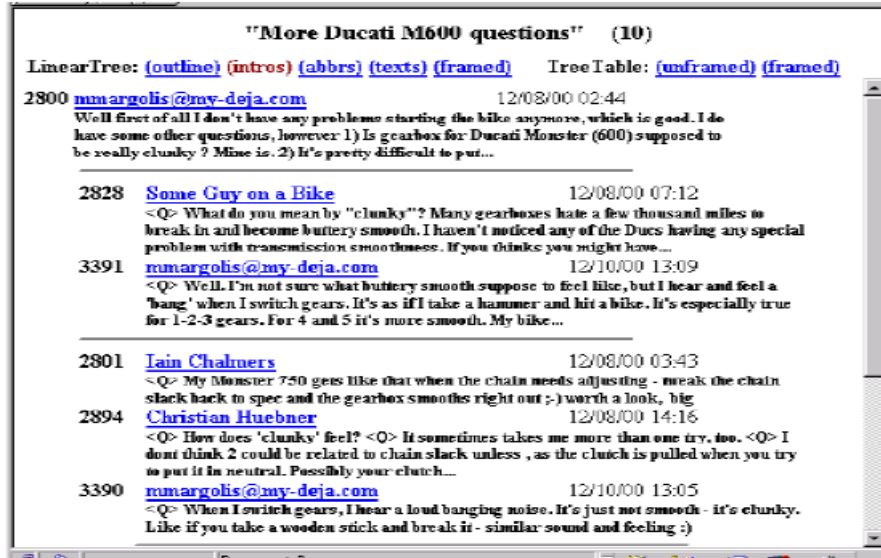


Figure 6.7: Narrow Tree for a conversation thread

cilitating text embedding (refer to Fig. 6.7). In the narrow tree representation, the depth-first conversation ordering is shown not by simply indenting each message under its predecessor, but by an alternative method that allows a narrower representation, and one more suited to text embedding. Briefly, a message is indented only if the predecessor has more than one response. In that case each response is indented and preceded by a dividing line. This representation is less satisfactory for bushy threads and it is hard to maintain context when there are frequent shifts in the relevant predecessors (signalled by the lack of indents).

Focus:	(col)	(undo)	Read:	(yes)	Size:	(focus)	(child)	(small)	(mid)
Extract:			Tree:	(frontup)	(fromfocus)				
1	2	3	4	5	6	7	8	9	10
11	12	13							
5918 Y:	Anyone have any pics or come with either of these brands?								
6065 Melander:	<Q> Can't say anything about Gerbing, but I've been pretty unimpressed with my Widder vest. The vest itself is fine, but the power cord connecting from the vest to the bike is inexcusably fragile. The wires break regularly where they...	6618 Ranzano:							
6473	6140 MacDonald: I have heard that the electric's of the Gerbings are parallel connections and the Widder are series connected. If one of the Widder gloves should fail, then the other will not work. The Gerbings will continue to work should one fail. I have both the Widder chaps and two of their vests with collars (one for passenger). Have served well and compress easily to store. Also, been using their electronic control with the coiled cord to keep wires out of the way. Their coiled cord is very smart. I find the Gerbing's cords to be a smaller wire gauge size - almost like a lamp cord (vinyl and not rubber?). Widder's sizes are snug so should be, but you may want next size up. May be better to give them your gut measurement instead of chest if your prone to eating a lot. Especially if you're a Wing rider. 2) The chaps stop the wind and provide noticeable heat - although not a lot. Gerbings supplies more at more draw on your electrical. I don't use the electric gloves (wiring is a pain to put on) as I have installed electric grips on bike.								
7302	6308 Onetumb: Even the cord on my electronic control (not the on/off switch kind) is nothing like a champ after 2 hard seasons of use. [MacDonald] I find the Gerbing's cords to be a smaller wire gauge size - almost like a lamp cord (vinyl and not rubber?). Yep. [MacDonald] Widder's sizes are snug as should be, but you may want next size up. May be better to give them your gut...	6342 N2BKK: I have the Gerbings Heated Jacket liner and I am very happy with it. But, as happy as I am, I am even more pleased with Gerbings as a company. When I received my liner it was no right in spots. I returned it for alterations and I got it back quickly. The alterations were done for free and Customer Service was GREAT! This is my 2nd season with it and hopefully many more to come. Wally	6472 Bike: It depends on what you get. The gloves are in series with each other. Everything else runs in parallel. I'm uncertain how the gloves wire up in relation to the Arm Chaps in the new System vests. [MacDonald] If one of the Widder gloves should fail, then the other will not work. The Gerbings will continue to work should one fail. What difference would it make? [MacDonald] Widder's sizes are snug as...						

Figure 6.8: Tree Table for a conversation thread where each cell represents a conversation session

Tree table [36] is a 2D tabular representation of the sub-conversations. In the tree table representation (see Fig. 6.8), cells represent nodes of the thread tree, and each cell in row i exactly spans the cells representing its children in the next row $i + 1$. Each column of the table represents a single path from the root of the thread tree to a leaf. The columns of the tree table represent the synchronous sub-conversations. An adapted focus+context approach is used to select either sub-trees or columns as foci. This representation provides efficient reading in both depth-

first and breadth-first direction by allowing either sub-trees or individual columns to be selected as focus, and to be expanded to different degrees. This enables the user to concentrate on a part of the conversation rather than the entire thread.

6.3 Chapter Summary

In this chapter, we have presented our conclusions based on the user study and have discussed a number of ways we could improve our interface in the future. We have explored the possibilities for better visualization of existing components like Marker Bars, Summary View and tags for the transcript using better color encoding, glyphs and details-on-demand views. We have also discussed extensions to several components like the Tag Selection Settings, Keyword Search Box and Entity View which would add more flexibility to their functionality. We have also skimmed a number of new possibilities like clustering the entities to reduce cognitive load and alternatives for displaying non-linear conversation threads. Although in our thesis and the proposed future work, in general, we have focused on visualization aspects of the conversation and the knowledge concepts related to it, there is much scope for improvement by developing more intelligent classifiers and a more task-oriented abstractive summary generating application.

Bibliography

- [1] Google search engine. → pages 98
- [2] Article on tag cloud, wikipedia. → pages 11
- [3] Color brewer application. → pages 48
- [4] Google mail. → pages 100
- [5] Timesketch, 2000. → pages 16
- [6] C. Albrecht-Buehler, B. Watson, and D. A. Shamma. TextPool: Visualizing Live Text Streams. In *Proceedings of the IEEE Symposium on Information Visualization, Washington, D.C., USA, 2004*. → pages 12
- [7] D. Angus, A. Smith, and J. Wiles. Conceptual Recurrence Plots: Revealing Patterns in Human Discourse. In *IEEE transactions on Visualization and Computer Graphics, Volume 17 Issue 12*, pages 1–14, 2011. → pages 15, 98
- [8] S. Björk and J. Redström. Window frames as areas for information visualization. In *Proceedings of the second Nordic conference on Human-computer interaction*, pages 247–250, 2002. → pages 25, 96
- [9] M.-M. Bouamrane and S. Laz. Navigating Multimodal Meeting Recordings with the Meeting Miner. In *Proceedings of FQAS*, pages 356–367, 2006. → pages 8, 38
- [10] G. Carenini and G. Murray. Visual Structured Summaries of Human Conversations. In *Proc. of IVITA 2010, HongKong, China*, pages 41–44, 2010. → pages 3, 31, 33, 94
- [11] G. Carenini, R. Ng, and X. Zhou. Summarizing Email Conversations with Clue Words. In *Proceedings of ACM WWW 2007*, pages 91–100, 2007. → pages 1

- [12] G. Carenini, G. Murray, and R. Ng. *Methods for Mining and Summarizing Text Conversations*. Morgan & Claypool Publishers, first edition, 2011. → pages 27, 30
- [13] J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, W. Kraaij, M. Kronenthal, G. Lathoud, M. Lincoln, A. Lisowska, I. McCowan, W. Post, D. Reidsma, and P. Wellner. The AMI Meeting Corpus: A Pre-Announcement. In *Proc. of MLMI 2005, Edinburgh, UK*, pages 28–39, 2005. → pages 8, 30
- [14] K. W. Church and J. I. Helfman. Dotplot: A Program for Exploring Self-Similarity in Millions of Lines of Text and Code. In *Journal of Computational and Graphical Statistics, Volume 2 No. 2 Jun*, pages 153–174, 1993. → pages 15
- [15] C. Collins, S. Carpendale, and G. Penn. DocuBurst: Visualizing Document Content using Language Structure. In *Computer Graphics Forum, Volume 28, Issue 3*, pages 1039–1046, 2009. → pages 23
- [16] R. Dachsel and M. Frisch. Mambo : A Facet-based Zoomable Music Browser. In *Proceedings of the 6th international conference on Mobile and Ubiquitous Multimedia*, pages 110–117, 2007. → pages 3, 7, 96
- [17] A. Don, E. Zheleva, M. Gregory, S. Tarkan, L. Auvil, T. Clement, B. Shneiderman, and C. Plaisant. Discovering interesting usage patterns in text collections: integrating text mining with visualization. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 213–222, 2007. → pages 18
- [18] F. B. V. Frank van Ham, Martin Wattenberg. Mapping Text with Phrase Nets. In *IEEE Transactions on Visualization and Computer Graphics, Volume 15, Issue 6*, pages 1169–1176, 2009. → pages 21
- [19] M. Galley. A Skip-Chain Conditional Random Field for Ranking Meeting Utterances by Importance. In *Proceedings of EMNLP 2006*, pages 364–372, 2006. → pages 1
- [20] W. Geyer, A. J. Witt, E. Wilcox, M. Muller, B. Kerr, B. Brownholtz, and D. R. Millen. Chat spaces. In *Proceedings of the 5th conference on Designing interactive systems: processes, practices, methods, and techniques*, pages 333–336, 2004. → pages 101, 102, 103

- [21] S. Havre, E. Hetzler, P. Whitney, and L. Nowell. ThemeRiver: visualizing thematic changes in large document collections. In *IEEE Transactions on Visualization and Computer Graphics Volume: 8 Issue: 1 Jan/Mar*, pages 9–20, 2002. → pages 19, 95
- [22] L. He, E. Sanocki, A. Gupta, and J. Grudin. Auto-summarization of audio-video presentations. In *Proceedings of ACM MULTIMEDIA*, pages 489–498, 1999. → pages 2
- [23] W. C. Hill, J. D. Hollan, D. Wroblewski, and T. McCandless. Edit Wear and Read Wear. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 3–9, 1992. → pages 10
- [24] P.-Y. Hsueh and J. D. Moore. Improving Meeting Summarization by Focusing on User Needs : A Task-Oriented Evaluation. In *Proceedings of the 14th international conference on Intelligent user interfaces*, pages 17–26, 2009. → pages 7, 8
- [25] Indratmo, J. Vassileva, and C. Gutwin. Exploring blog archives with interactive visualization. In *Proceedings of the working conference on Advanced Visual Interfaces*, pages 39–46, 2008. → pages 10, 15
- [26] K. S. Jones and J. Galliers. *Evaluating Natural Language Processing Systems: An Analysis and Review*. Springer, 1995. → pages 27
- [27] D. Jurafsky and J. H. Martin. *An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Prentice Hall, second edition, 2009. → pages 9
- [28] B. Kerr and E. Wilcox. Designing remail: reinventing the email client through innovation and integration. In *CHI '04 extended abstracts on Human factors in computing systems*, pages 837–852, 2004. → pages 100, 101, 102
- [29] Z. Kozareva and E. Hovy. A semi-supervised method to learn and construct taxonomies using the web. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1110–1118, 2010. → pages 26, 98
- [30] K. McKeown, J. Hirschberg, M. Galley, and S. Maskey. From Text to Speech Summarization. In *Proc. of ICASSP*, pages 997–1000, 2005. → pages 2
- [31] P. Moody and D. Fisher. Studies of automated collection of email records. Technical report, University of Irvine, 2002. → pages 102

- [32] T. Munzner. *Information Visualization: Principles, Methods, and Practice*. first draft edition, 2011. → pages 35, 48, 49
- [33] G. Murray and G. Carenini. Interpretation and Transformation for Abstracting Conversations. In *North American ACL, Los Angeles, CA, USA*, 2010. → pages 33, 53, 54
- [34] G. Murray, T. Kleinbauer, P. Poller, S. Renals, T. Becker, and J. Kilgour. Extrinsic Summarization Evaluation: A Decision Audit Task. In *Proceedings of MLMI 2008*, pages 349–361, 2008. → pages 2
- [35] G. Murray, G. Carenini, and R. Ng. Generating Abstracts of Meeting Conversations: A User Study. In *Proceedings of the 6th International Natural Language Generation Conference*, pages 105–113, 2010. → pages 27, 29, 37, 54
- [36] P. S. Newman. Exploring discussion lists: steps and directions. In *Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries*, pages 126–134, 2002. → pages 105, 107
- [37] W. B. Paley. TextArc: Showing Word Frequency and Distribution in Text. In *InfoVis Poster Compendium*, 2002. → pages 13
- [38] V. Pascual-Cid and A. Kaltenbrunner. Exploring asynchronous online discussion through hierarchical visualization. In *Proceedings of 13th International Conference Information Visualisation*, pages 191–196, 2009. → pages 100, 102, 104, 105
- [39] O. Rambow, L. Shrestha, J. Chen, and C. Lauridsen. Summarizing Email Threads. In *Proceedings of HLT-NAACL 2004: Short Papers*, pages 105–108, 2004. → pages 1
- [40] S. Rashid and G. Carenini. An Ontology-based Visual Interface for Browsing and Summarizing Conversations. In *Proc. of VISSW workshop, Palo Alto, California, USA*, 2011. → pages 32
- [41] M. Sedlmair, C. Bernhold, D. Herrscher, S. Boring, and A. Butz. MostVis: An Interactive Visualizing Supporting Automotive Engineers in MOST Catalog Exploration. In *Proc. of IV, New Jersey, USA*, pages 173–182, 2009. → pages 10, 36
- [42] G. Smith, M. Czerwinski, B. Meyers, D. Robbins, G. Robertson, and D. S. Tan. FacetMap: A Scalable Search and Browse Visualization. In *IEEE*

Transactions on Visualization and Computer Graphics, Volume: 12 Issue:5, pages 797–804, 2006. → pages 9

- [43] M. A. Smith and A. T. Fiore. Visualization components for persistent conversations. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 136–143, 2001. → pages 100, 102
- [44] G. Tur, A. Stolcke, L. Voss, S. Peters, D. Hakkani-Tur, J. Dowding, B. Favre, R. Fernandez, M. Frampton, M. Frandsen, C. Frederickson, M. Graciarena, D. Kintzing, K. Leveque, S. Mason, J. Niekrasz, M. Purver, K. Riedhammer, E. Shriberg, J. Tien, D. Vergyri, and F. Yang. The CALO Meeting Assistant System. In *IEEE Transactions on Audio, Speech, and Language Processing, Volume: 18 Issue:6*, pages 1601–1611, 2010. → pages 9
- [45] G. D. Venolia and C. Neustaedter. Understanding sequence and reply relationships within email conversations: a mixed-model visualization. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 361–368, 2003. → pages 100, 101, 103
- [46] F. B. Viégas, S. Golder, and J. Donath. Visualizing email content: portraying relationships from conversational histories. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 979–988, 2006. → pages 13
- [47] F. B. Viegas, M. Wattenberg, and J. Feinberg. Participatory Visualization with Wordle. In *IEEE transactions on Visualization and Computer Graphics, Volume 15 Issue 6*, pages 1137–1144, 2009. → pages 11
- [48] C. Ware. *Information Visualization: Perception for Design*. Morgan Kaufmann Publishers, second edition, 2004. → pages 48
- [49] M. Wattenberg. Arc Diagrams: Visualizing Structure in Strings. In *IEEE Symposium on Information Visualization*, pages 110–116, 2002. → pages 16
- [50] M. Wattenberg and F. B. Viegas. The Word Tree, an Interactive Visual Concordance. In *IEEE Transactions on Visulaization and Computer Graphics Volume 14, Issue 6*, pages 1221–1228, 2008. → pages 20
- [51] M. Weiss-Lijn, J. T. McDonnell, and L. James. Supporting Document Use Through Interactive Visualization of Metadata. In <http://vw.indiana.edu/visual01/weiss-lijn-et-al.pdf>, 2001. → pages 24

- [52] P. Wellner, M. Flynn, and M. Guillelot. Browsing Recorded Meetings with Ferret. In *Proc. of MLMI 2004, Martigny, Switzerland*, pages 12–21, 2004. → pages 8, 24, 38
- [53] W. Willett, J. Heer, and M. Agrawala. Scented Widgets: Improving Navigation Cues with Embedded Visualizations. In *IEEE Transactions on Visualization and Computer Graphics, Volume: 13 Issue: 6*, pages 1129–1136, 2007. → pages 24, 36
- [54] H. Yang and J. Callan. Ontology generation for large email collections. In *Proceedings of the 2008 international conference on Digital government research*, pages 254–261, 2008. → pages 26, 99
- [55] K.-P. Yee, K. Swearingen, K. Li, and M. Hearst. Faceted metadata for image search and browsing. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 401–408, 2003. → pages 3, 7, 96
- [56] L. Zhou and E. Hovy. Digesting Virtual “Geek” Culture: The Summarization of Technical Internet Relay Chats. In *Proceedings of ACL 2005*, pages 298–305, 2005. → pages 2
- [57] X. Zhu and G. Penn. Summarization of Spontaneous Conversations. In *Proceedings of Interspeech 2006*, pages 1531–1534, 2006. → pages 2

Appendix A

Supporting Materials

A.1 Consent Form



The University of British Columbia

Department of Computer Science
2366 Main Mall
Vancouver, B.C., V6T 1Z4

Jan, 2012

Subject Consent Form

**Project Title: A Visual Interface for Browsing and Summarizing
Conversations**

Principal Investigators

Shama Rashid
Department of Computer Science,
University of British Columbia
(778) 998 6174

Dr. Giuseppe Carenini
Associate Professor,
Department of Computer Science,
University of British Columbia
(604) 822 5109

Student Investigators

Anders Linn
Department of Cognitive Systems,
University of British Columbia
(778) 938 3751

Project Purpose and Procedures

The purpose of this project is to interactively browse a human conversation and to generate summary for it using a visual interface. You will be given further instructions on the tasks. If you are unsure about any instruction, please do not hesitate to ask the investigator.

Time Commitment

The session will take approximately 2 hours.

Confidentiality

The identities of all people who participate will remain anonymous and will be kept confidential. All data from individual participants will be coded so that their anonymity will be protected in any reports, research papers, thesis documents, and presentations that result from this work.

Remuneration/Compensation

You will receive a \$15/hr honorarium for your participation in this project.

Contact for Information about the Rights of Research Subjects

If you have any concerns about your treatment or rights as a research subject, you

may contact the Research Subject Information Line in the UBC Office of Research Services at 604 822 8598.

Consent

We intend for your participation in this project to be pleasant and stress-free. Your participation is entirely voluntary and you may refuse to participate or withdraw from the study at any time. The experimenter will answer any questions you have about the instructions or the procedure of this study.

Your signature below indicates that you have received a copy of this consent form for your own records.

Your signature indicates that you consent to participate in this project. You do not waive any legal rights by signing this consent form.

I, _____, agree to participate in the project as outlined above.
My participation in this project is voluntary and I understand that I may withdraw
at any time.

Participant's Signature

Date _____

Investigator's Signature

Date _____

A.2 Pre-questionnaire

Participant No. :

Participants Gender: F / M

Date:

Age:

Education:

- None

- Some highschool
- Highschool diploma
- Some university
- Bachelors degree
- Masters degree
- Ph.D

How would you rate your English proficiency level?

- Very poor
- Poor
- Average
- Good
- Excellent

On a scale of 1-10 (where 1 means you rarely use computers and 10 means you could be considered an expert computer user), how comfortable are you with computers?

Have you ever taken a computer science class? If so, which ones?

Do you have any vision problems (color blindness, corrected vision, etc.)?

How many hours, on average, do you spend daily on a computer?

A.3 User Study Tasks

User Study on a Visual Interface for Browsing and Summarizing Conversations

Introduction:

In this experiment you will be using a visual interface to browse a human conversation in order to answer questions about discussions that took place during the conversation and to automatically generate summary for it.

The user study will be conducted in 3 stages: a) tutorial session, b) practice session, and c) experiment session. For each session, write down your answers for the tasks assigned in the text area provided under the tab Answer Pad on the interface and remember to save your answer at the end of the session.

Now lets start with the Tutorial Session!

Tutorial Session:

In this session, the experimenter will introduce you to the interface and show you how to use its different features to analyze a sample conversation using the example tasks given below. Please do not hesitate to ask the experimenter if you need any further clarification.

Example Tasks:

1. Can you guess the subject of the conversations?
2. What kind of coat does Susan want to buy? Why did she not want a red coat or a tri-climate one?
3. What does Susan want for lunch? Did she consider other options? If so, what were the other choices?

Take as much time as you deem necessary to get accustomed to the interface. Once you feel comfortable using the interface please notify the experimenter that you are

ready to move on to next stage, the Practice Session.

PS:

Please note that the Sentence Type and Topic tags on the interface were done automatically and may not be complete or fully accurate. You can copy a line directly from the transcript view to the Answer Pad using CTRL+C and CTRL+V keyboard shortcuts.

Practice Session:

In this session, you will work on a set of sample tasks involving a conversation between two friends, Betty and Ronnie, as they plan how to spend their time in the evening.

Instead of reading through the entire transcript, we suggest that you use the features of the interface as much as possible to get to your answer more easily. You will be working primarily on your own, but feel free to ask the experimenter for any help or clarification required. Take as much time as you feel necessary to complete the tasks.

Tasks:

1. What types of movies did Betty and Ronnie consider watching? What movie did they finally decide to watch? Why did they reject the other options?
2. What party are they planning to go to? Why is Betty worried about preparing for the party?

Once you have finished the tasks, please notify the experimenter.

Experiment Session:

In this session, you will be working on a number of tasks for a series of meeting conversations where a project group is discussing the design of a remote control. You may need to browse all 4 meeting sessions to answer some of the questions or the answer may be presented in just one of the meetings. Reading through the entire transcript is very timing consuming, so we suggest using the features of the

interface as much as possible in order to find answers quickly. You will be working on your own for this session; however, feel free to ask the experimenter for help if you have any questions related to the interface. You have one hour to complete the tasks.

Consider a scenario where you have recently joined a product designing company, Real Reactions. The company has marketed a newly designed remote control last year and the sales report for the first quarter has just come in. Based on the sales statistics and the customer reviews collected online, the Marketing Division is trying to figure out how well-accepted and profitable this new product is. They have appointed you to re-assess the design decisions for the device and have provided you with the original meeting transcripts of the product design team which consisted of a Project Manager, a Marketing Expert, an Industrial Designer and a User Interface Expert. The design team held a series of four meetings: 1) kickoff, 2) conceptual design, 3) detailed design, and 4) evaluation, resulting in the current design. Your task is to answer the queries posed by the Marketing Division. For each task, also mention the line numbers of the transcripts that you used to get at your reply (for example, if you have used line number 2012 of meeting transcript 4 then write down 4.2012 in the answer pad; you must be clear about which meeting in the series the information appeared in and on which line).

Tasks:

1. The Marketing Division wants to analyze the quarterly sales report to find out whether the remote control launched is in keeping with the forecast by the project team. What was the target cost of a remote control unit and the target final consumer price? What was the total amount the company was targeting to earn from this product? Which team member mentioned the target amount first?
2. A recent market survey revealed that battery life was an important feature considered by customers while buying a remote control. What are the options the design group consider about power of the remote? What was the final decision? Who proposed that solution?

3. One of the online customer reviews for the product read A scroll-wheel like Apples ipod would have been cool for channel surfing! Did the project group consider this option? If so, why was it discarded? What did they decide on in the end as the main way of interaction?.
4. People in the Marketing Division believe that the remote control does not sufficiently promote the corporate image. How did the design group decide to incorporate the companys corporate image with the design of the remote control device?
5. Much of the user feedback collected online stated the need for an LCD screen based menu. Was the idea discussed at all? If it was considered then why was the idea rejected in the final design?

A.4 Post-questionnaire

Participant No. :

Date:

For each statement in the following section, indicate how strongly you agree or disagree with the statement by circling the most relevant number (for example, 1=disagree strongly and 5=agree strongly)

1. I found the conversation browser intuitive and easy to use.

disagree strongly 1 - 2 - 3 - 4 - 5 agree strongly

2. I was able to find all of the information I needed.

disagree strongly 1 - 2 - 3 - 4 - 5 agree strongly

3. I was able to find the relevant information quickly and efficiently.

disagree strongly 1 - 2 - 3 - 4 - 5 agree strongly

4. I feel that I completed the task in its entirety.

agree strongly 1 - 2 - 3 - 4 - 5 disagree strongly

5. The task required a great deal of effort.

agree strongly 1 - 2 - 3 - 4 - 5 disagree strongly

6. I felt I was working under pressure.

disagree strongly 1 - 2 - 3 - 4 - 5 agree strongly

7. I had the necessary tools to complete the task efficiently.

agree strongly 1 - 2 - 3 - 4 - 5 disagree strongly

8. I would have liked the conversation browser to have contained additional information about the conversations.

disagree strongly 1 - 2 - 3 - 4 - 5 agree strongly

9. The interface quickly reflected the changes caused by interaction (changes caused when you select or unselect tags etc.)

disagree strongly 1 - 2 - 3 - 4 - 5 agree strongly

In the following section, please answer the questions with a short response of 1-3 sentences.

10. How useful did you find the conversation summaries and the linking of summary sentences to the transcript?

11. How useful did you find the sentence speaker and sentence type tags in the Ontology View?
12. How useful did you find the Topic View?
13. How useful was the Tag Selection Settings?
14. How useful were the marker bars and the tool tip texts (the text shown when you place the mouse pointer over a marker)?
15. What information would you have liked to have that wasn't available?
16. How accurate and complete was the list of topics in the Topic View?
17. How accurate and complete was the sentence type listing (Decision, Problem etc.)?
18. Do you have any comments on how the interface could be improved?

A.5 Marking Scheme for the User Study Experiment Session and Instructions to the Judges

Marking Scheme:

Q1. Full marks: 2

production cost 12.50 = 0.5 marks

selling price 25.00 = 0.5 marks

Target profit 15 million euro = 0.5 marks

PM introduced the project finances = 0.5 marks

Q2. Full marks: 3

4 options: 0.5 marks each

a) Lithium ion/long lasting/rechargeable batteries

b) solar power

c) double A

d) kinetic batteries

Final choice kinetic battery = 0.5 marks

Kinetic battery proposed by User Interface Expert or UIE = 0.5 marks

Q3. Full marks: 3

Yes, they discussed ipod like scroll wheel = 1 mark

Decided to use push buttons instead = 1 mark

They rejected the scroll wheel because its a) expensive and b) harder to control =
1 mark *if either reason mentioned*

Q4. Full marks: 2

corporate logo or R_R_ (the name of the company is Real Reactions) = 1 mark

color yellow or corporate color = 1 mark

Q5. Full marks: 2

Yes, the idea of LCD based menu was discussed = 1 mark

discarded because of a) cost since would require a more expensive chip and b)
readability issues (too small and user will have to keep switching from looking at
the remote and the TV) = 1 mark *if either reason mentioned*

Instructions for the Judges:

Please also highlight or mark the copied sentences that you consider relevant to the task. Some of the participants did not consolidate the copied transcript sentences as they were instructed. Even if a participant has not answered the question in her own words, if the relevant information can be found in the copied lines, give them the marks. To be fair to participants who have tried to write down the answers in their own words, if they did not get full marks, please have a second look at the

copied lines to see whether they had at least identified the relevant information and adjust their marks accordingly.

A.6 Pilot Study

User Study on a Visual Interface for Browsing and Summarizing Conversations

Introduction:

The purpose of this project is to test an interface designed to help browse human conversation and to generate summary for it.

Please take as much time as you deem necessary to get accustomed to the interface. Please notify the investigator when you feel ready to perform the task.

Stage 1:

Allotted Time:

You are allowed 45 minutes to complete this task.

Task Instructions:

The group discussed the issue of separating the commonly-used functions of the remote control from the rarely-used functions of the remote control. What was the final decision on this design issue? Please write a short summary (1-2 paragraphs) describing the final decision, any alternatives the participants considered, the reasoning for and against any alternatives (including why each was ultimately rejected), and in which meetings (Meeting 1, 2, 3, or 4) the relevant decisions were made.

Attention:

The annotation for the utterance types and the entities have been done automatically. So, they may not be fully accurate.

Stage 2:

Task Instruction:

You will receive further instructions on Stage 2 after you have completed Stage 1.

Thank you for participating in this user study!

P.S. - Please remember to sign the consent form if you haven't done so already.