

# A Visual Interface for Browsing and Summarizing Conversations

Shama Rashid

Department of Computer Science  
University of British Columbia  
shama@cs.ubc.ca

## ABSTRACT

In this paper I present a visual interface for generating focused summaries for human conversations. My interface provides the user a way to quickly explore a conversation to get an overview of its content and to interactively identify informative sentences as potential components of the summary. The candidate sentences are selected based on visual cues provided by mapping to an ontology containing nodes for the speakers of the conversation, sentence-level features like Dialog Acts (in short DAs, like decision, action, problem etc.) properties, and entities referred to in the conversation. The scope of this course project is to enhance the usability of the interface based on information visualization principles. I shall specifically address the issues of a better visual representation for the ontology information and easier exploration of the conversation transcript.

## TERMINOLOGY

Through out the scope of this project the terms ‘utterance’ and ‘sentence’ will be used interchangeably to indicate a segment of the conversation that can express a meaning independent of other segments. The people involved in the conversation will be referred to as ‘participants’ or ‘speakers’. ‘Entities’ are noun phrases that appear in the conversation with a mid range document frequency. ‘Dialog Acts’ are sentence level features that specify whether a sentence expresses positive or negative opinions, a decision, an action or a problem.

## DOMAIN, TASK AND DATASET

In our daily lives, we have conversations with people in many different modalities - emails, meetings, telephone, videoconferencing, instant messaging, blogs, forums, etc. The Web has significantly increased the volume and the complexity of the conversational data generated through our day to day communication. The ontology containing nodes for speakers, dialogue acts, and a list of entities referred to in the conversation is derived using classifiers and parsers based on generic features. So, the interface could be used to explore

any conversational data provided (offline) in a text format.

With the high volume of conversational data being generated every day, summarization can be beneficial by creating concise overviews that aid quick access to the data. Extractive summary generated by selecting and concatenating the most informative sentences is the dominant approach in Natural Language Processing (NLP), since it can be considered a simple binary classification task of whether to include a particular sentence in the summary or not. However, the resultant summary is often unsatisfactory for users due to lack of coherency. The alternative approach is abstractive summary, which is generated by extracting and aggregating information from the conversation. This approach requires a natural language generation component and is preferred by users for coherency but lacks the context in which each of the abstractive summary sentences was generated. The objective of my interface is to provide a Visual Structured Summary that gives 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.

I’ve been working mainly on the AMI meeting corpus [3]. 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. Currently the interface can be used to analyze a single transcript, but in the future I’ll extend it to analyze the whole series of transcripts together. I am currently extending the interface to other modes of conversation like emails, blogs, chats etc. and have also been working with the BC3 email corpus [13] and IBM technical blogs crawled off the web.

An example for mapping a sentence to the ontology can be found in figure 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

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, entity). Additionally,

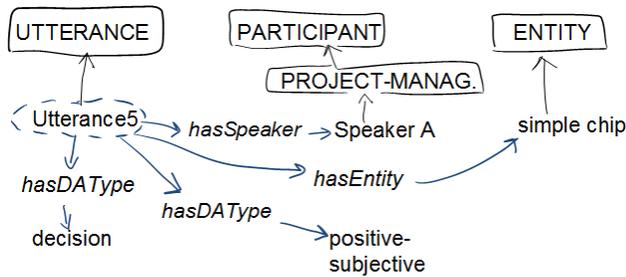


Figure 1. Mapping a sentence to an ontology example

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, the attribute 'ProjectManager' (say) would have a value 'yes' but the other speaker attribute columns would contain a 'no' for this particular item.

### PERSONAL EXPERTISE

I've been working on this project under an RA-ship for slightly longer than a year, since September 2010, as my M.Sc. thesis under the supervision of Dr. Giuseppe Carenini. I've also received valuable feedback on the interface from Dr. Raymond Ng and Dr. Gabriel Murray over the length of that time period. I had started with the interface idea described in Section 'Initial Design' and have developed a fully-functional interface as presented in Section 'Current Design'. The current design has been developed through a number of iterations, alternating between reviewing related literature on conversation browsers and visualization of document collections and reapplying or extending ideas presented in those papers in my implementation. The details of the relevant projects can be found in the 'Related Works' Section. So far, I've conducted one pilot study to evaluate the design approach and assess the effectiveness of different components of the current interface.

### INITIAL DESIGN

The details on an initial interface to aid creating visual structured summaries of conversations can be found in [2]. This 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 properties of the utterance/sentence such as whether it was expressing a decision, a problem etc i.e. the DAs. 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 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.

### CURRENT DESIGN

The current design (see Fig 3) addresses several limitations of the initial prototype presented in the previous section. It consists of three integrated views - the Ontology View, the Transcript View and the Summary View. The details on the display and interaction design of the current implementation can be found in [9]. However, I'm reiterating a few problems with the initial design from information visualization point of view and the solution proposed in the current design below.

The accuracy 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. In addition, the solution 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 in such a situation. The actual situation is far worse as a sentence can be classified as multiple DA types simultaneously. Another conceptual problem with this solution was using colors to show two different categorical attributes (speaker, DA type) when the color map for categorical + categorical attributes are perceptually inseparable. The solution I came up with in the current design is, instead of using color, I've added a column (named Tags) to the left of the transcript utterances, in which the (selected) mapping to the (knowledge concepts within) ontology of the corresponding utterance can be displayed.

The ontology representation now also includes entities mentioned in the conversation. Searching the conversation using a particular keyword can only be used when 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 browsing of the conversation. In addition, the entities also provides the user with a quick overview of the content of the whole conversation without browsing the transcript. I have also included count within parentheses beside the labels on the ontology tree as information scent. For leaf nodes, the count indicates how many sentences were mapped to this node and imply its relevance for the summary; for non-leaf nodes this is just the sum of all its descendant leaf node count.

To take advantage of visual popout, I've decided to use icons associated with the ontology concepts instead of text labels as representation of tags in the Tags column. A word can have different meanings in different contexts, that's why instead of showing the entity tags in the Tags column, I'm highlighting them in bold blue font within the transcript sen-

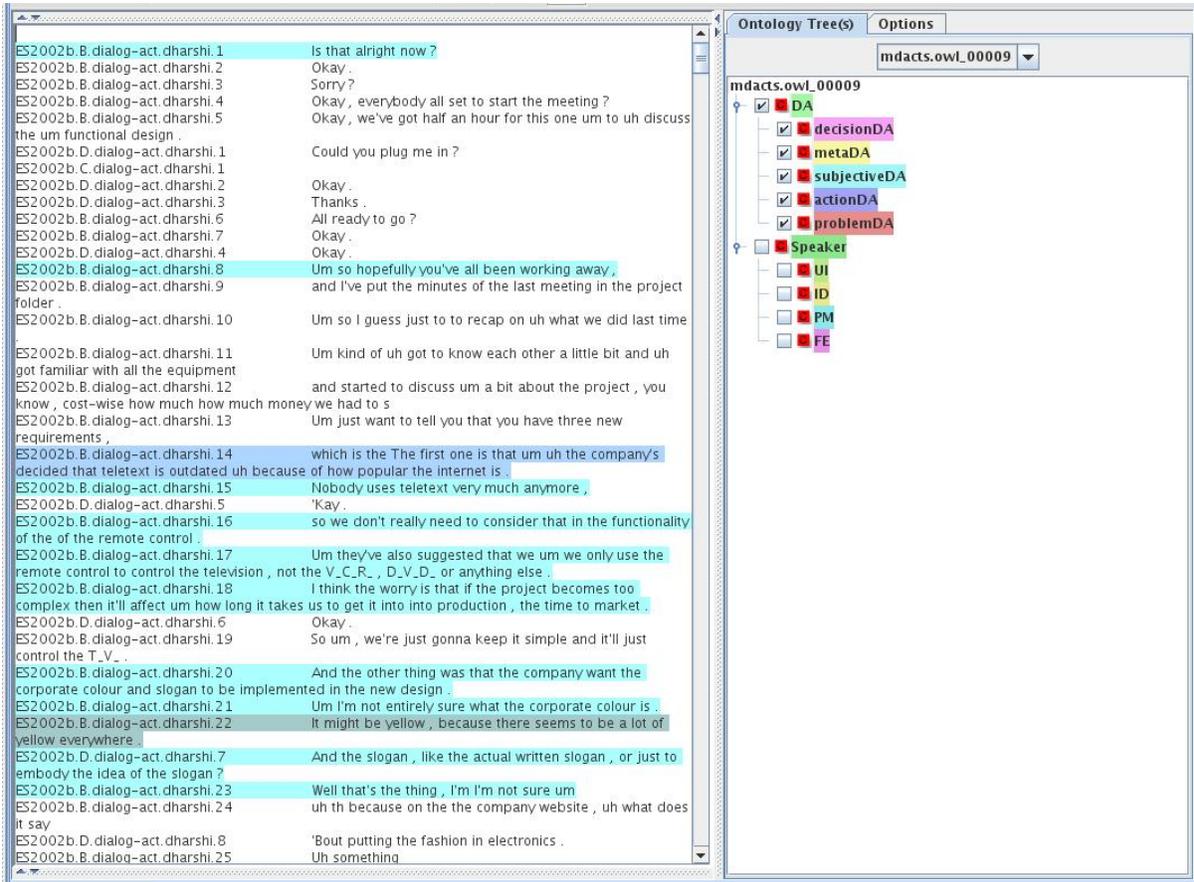


Figure 2. Initial interface

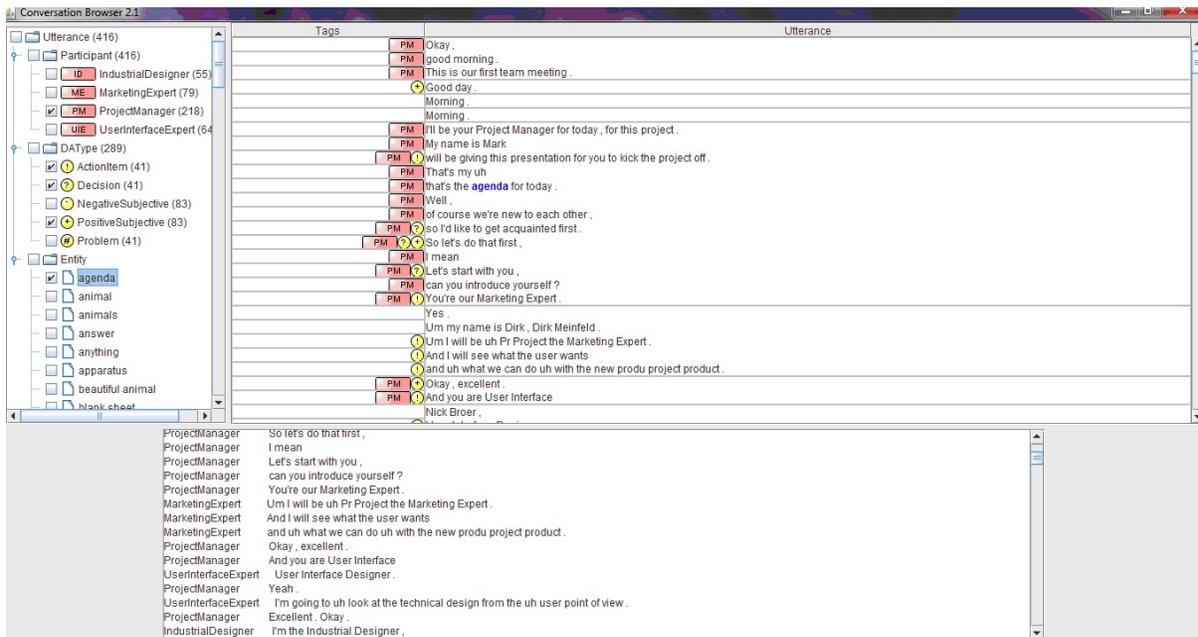


Figure 3. Current prototype with the Ontology View (right), the Transcript View (middle) and the Summary View (bottom)

tences.

I've 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) are grouped using containment within a larger grid box.

The summary view works as a filtered view of the sentences that can be 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 also an important addition to the interface since 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 [8] 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*).

## PROPOSED SOLUTION

As a part of this course project I am proposing several modifications to the current design to enhance the usability of the interface. Different aspects of the extended solution are discussed below.

### Separation of Entities from DA type and Speaker nodes

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 of this information scent, an easy extension would be to scale the font size of the labels i.e. using a larger font for nodes with larger counts to make them stand out more (figure 5 left). 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 I'll have 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 is not possible as it would assign the maximum font size for most of the nodes 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, I am proposing to display the entities in a tag cloud format separate from the Ontology view. This would allow me to use two different scaling functions for the labels in the two different views. The details of this new view containing entities are discussed below.

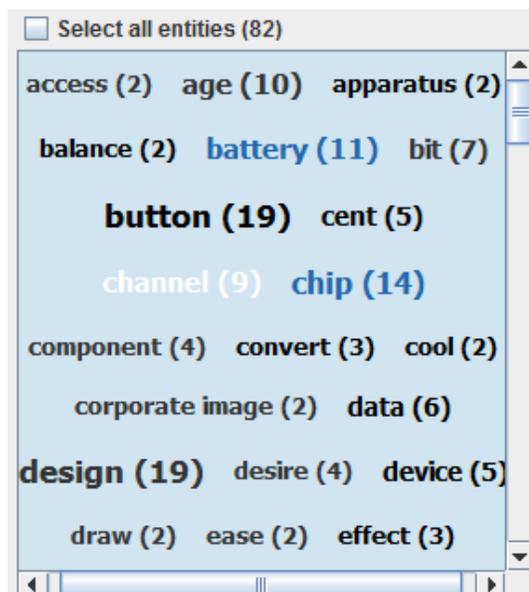


Figure 4. Proposed entity cloud

### Entity Cloud

The entity cloud view will be 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 will be listed alphabetically in a sequential line-by-line layout. The font size of each entity will be scaled using a linear scaling function. It will be possible to select multiple entities simultaneously from the list and I'll use different colors to differentiate entities that are selected from the unselected ones (or for candidates for mouse hover).

Another possibility is to sort the entities according to frequency of occurrence in ascending or descending order. I'm planning to achieve the same effect by providing a range slider where the user will be able to specify a minimum and a maximum count for the list of entities. This would fade out the entities falling outside the range selected and thus narrow down the search scope.

### Redesign Icons

In the current design I'm using pink rectangles with abbreviation of the speaker name as icons for the Speaker nodes, and yellow circles with characters like '+', '-' etc. to indicate PositiveSubjectivism, NegativeSubjectivism etc. for

the DA type nodes. There is a strong domain convention to use red for negative sentiments and green for positive sentiments, which I'll follow for the redesigned icons and I'll use luminance to counter the limitation of using red-green hues for color-deficient users. I'm planning on taking advantage of the shape channel by making the icons for different leaf nodes of distinguishable shapes. So, instead of using a circle containing a '+' or '-' sign I'll use a green '+' or a red '-' shaped icon. For the other nodes, I'll have to search for appropriate representative shapes.

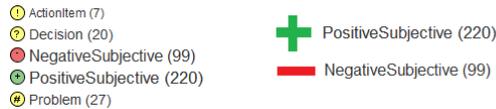


Figure 5. Current and proposed icons for DA types

### Marker Bar

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 normally inspection of the context sentences is mandatory to fully understand the relevance of the tagged sentence and sometimes the tagged sentences are widely spread out through out the conversation length. I'm proposing to add a marker bar (see figure 6) to the transcript view to reduce this scroll time. Clicking on a marker would auto-scroll the transcript and make the part of the transcript corresponding to the tag associated with that marker visible to the user. I'll be using different colored markers for the three broad categories of tags i.e. DA types, speakers and entities.

### Information Scent for the Summary Sentences

In the current design the extractive summary sentences are prefixed by the speaker of the sentence. This lacks the information why a sentence was selected to appear in the summary in the first place (unless the user was only exploring the Speaker subtree). I'll prefix the tags for the selected ontology concept that can be applied to the sentence to make the summary view self-contained without looking up the transcript. Also, I'll add the line number for the sentences in the conversation in the transcript view as well as in the summary view. This will provide 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 scent is even more important for abstractive summary approach since for this approach the sentences are aggregations of a set of transcript sentences and without the context information 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 uncertainty involved. For abstractive summary sentences, there will be a list of line numbers for the component transcript sentences instead of one as shown for the extractive summary approach in figure 7.

### Others

The entity cloud is a list of the noun phrases in the conversation extracted automatically, hence it cannot be taken as a complete enlistment of entities. Also, some adverb and verb forms are strong cues for conversational direction. For example, to find a decision taken in a conversation a user could lookup the sentences mapped to the 'Decision' DA type node or look for keywords like 'finally'. In addition to browsing the conversation in a structured way using the tag cloud and the ontology tree, I'll provide a keyword search option to the user to increase flexibility of search.

Often the information the user is searching for appears in the context of the tagged sentence instead of the tagged sentence itself. I'll provide the user a way to mark these sentences as important and also add markers on the marker bar for them so that the user may easily find them later to re-inspect them.

### SCENARIO OF USE

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 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 entities in the tag cloud and select entities like 'remote control' and other entities like 'button' or 'lcd' that seem related to remote control designs (Fig 4). This would highlight those entities in the transcript and markers along the marker bar would appear (Fig 6). At this point the user could employ two different approaches to inspect the sentences further; she could click on each marker and read the corresponding portion of the transcript, or she could look at the filtered sentences in the summary view 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. 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. To find what other topics were highly discussed the user might want to use the range slider provided for the entity cloud and set the range to a portion in the higher range of the scale. This would significantly reduce the number of entities 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.

### PROPOSED IMPLEMENTATION APPROACH

All data parsing, formatting and ontology generation in the current design has been done using Python scripts; the formatted files were stored offline to reduce data loading time. The current interface has been built using Java Swing and AWT components. I am also using a Java based framework, Jena, for querying the ontology. The natural language generation component used to derive the abstractive summary is based on Java and SimpleNLG api. Since the entire front-end application has been developed using Java, it is platform independent and free-source. For the extensions proposed for the interface within the scope of the project I'm planning to reuse freely available Java toolkits and apis as much as possible.

## MILESTONES AND SCHEDULE

Task	Tentative Date of Completion
Submit project proposal	28-Oct-11
Find a suitable package for the marker bar	31-Oct-11
Rephrase technical jargon, implement tag selection settings, find better icons for sentence types and speaker	02-Nov-11
Incorporate speaker as a turn parameter tag and use optional background coloring of transcript according to speakers	06-Nov-11
Implement entity tag cloud and entity frequency range slider	08-Nov-11
Include indicators in the summary view why a summary sentence was generated	10-Nov-11
Incorporate keyword search box and marker bar for the transcript view	15-Nov-11
Present update on course project	21-Nov-11
Conduct informal user study (and possibly go through for another iteration of design cycle)	08-Dec-11
Present course project	12-Dec-11
Submit final project paper	14-Dec-11

Table 1. Milestones and tentative schedule

## PREVIOUS WORK

(I am copying this section from [9])

In HCI and NLP different approaches have been proposed to support the browsing and summarization of data/documents with the aid of an interactive interface. Here, we focus on the ones that are more critical for our current and future work.

The idea of using an ontology to explore data in an orderly manner is not novel. For instance, the Flamenco [15] and the Mambo [4] systems make use of hierarchical faceted meta-data for browsing through image or music collections. In our approach we adopt similar techniques to support the exploration of conversations. More specifically, in Flamenco [15], 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. Another idea we have borrowed from the Flamenco and Mambo systems is to use summary paths to simplify the user interaction with the ontology. 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 are hidden). So, it may happen that the set of tags selected by the users is not fully visible. To address this problem, we are working on including a summary of the ontology node selection at the top of our interface, as it is done in Flamenco and Mambo.

An extractive approach for generating a decision-focused summary suitable for debriefing tasks has been proposed in [6]. 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 [6] 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 we are also exploring focused summarization by abstraction. The interface proposed in [6] also considers features that are specific to conversations about designing a new product (see AMI corpus [3]), 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 conversation we intend to do it in a general way, i.e., not being limited to conversations about designing a product.

The Ferret Meeting Browser [14] 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. In the future, we'll extend our interface to include an overview of the conversation integrating ideas from the following projects.

The Meeting Miner [1] 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 selected from a list of automatically generated keywords and topic. 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 [1] 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 [11] 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 the ontology tree of our interface are equivalent to Meeting Miner's keyword panel entries and we are currently listing the entities in alphabetical order; but a different ordering based on the count etc. may prove more helpful to the users.

The CALO meeting assistant [12] is used for capturing audio signals and optional handwriting recorded by digital pens for distributed meetings. 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. 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 abstractive or extractive summaries.

In iBlogVis [7], 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 shall employ in the future for our textual collage representation of conversation content. iBlogVis uses the idea of read wear [5], 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. Similarly, we are currently working to provide users an option to log the current ontology settings so that they can keep track of the combinations tried before.

MostVis [10] 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 is similar to our use of a short string representation or icon beside the ontology tree nodes. MostVis also has a history window with undo and redo button 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.

## REFERENCES

1. M.-M. Bouamrane and S. Laz. Navigating Multimodal Meeting Recordings with the Meeting Miner. In *Proc. of FQAS, Milan, Italy*, 2006.
2. G. Carenini and G. Murray. Visual Structured Summaries of Human Conversations. In *Proc. of IVITA 2010, HongKong, China*, pages 41–44, 2010.
3. 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.
4. R. Dachselt and M. Frisch. Mambo : A Facet-based Zoomable Music Browser. In *Proc. of MUM, Oulu, Finland*, 2007.
5. W. C. Hill, J. D. Hollan, D. Wroblewski, and T. McCandless. Edit Wear and Read Wear. In *Proc. of SIGCHI, Monterey, California, USA*, 1992.
6. P.-Y. Hsueh and J. D. Moore. Improving Meeting Summarization by Focusing on User Needs : A Task-Oriented Evaluation. In *Proc. of IUI, Florida, USA*, 2009.
7. Indratmo, J. Vassileva, and C. Gutwin. Exploring Blog Archives with Interactive Visualization. In *Proc. of AVI, Naples, Italy*, 2008.
8. G. Murray, G. Carenini, and R. Ng. Generating Abstracts of Meeting Conversations: A User Study. In *Proc. of INLG, Dublin, Ireland*, 2010.
9. 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.
10. 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 InfoVis09, New Jersey, USA*, 2009.

11. G. Smith, M. Czerwinski, B. Meyers, D. Robbins, G. Robertson, and D. S. Tan. FacetMap: A Scalable Search and Browse Visualization. In *Proc. of InfoVis, Baltimore, Maryland, USA, 2006*.
12. G. Tur, A. Stolcke, L. Voss, S. Peters, D. Hakkani-Tr, 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 Transaction on Audio, Speech and Language Processing*, 2010.
13. J. Ulrich, G. Murray, and G. Carenini. A Publicly Available Annotated Corpus for Supervised Email Summarization. In *Proc. of AAAI EMAIL-2008 Workshop, Chicago, USA, 2008*.
14. P. Wellner, M. Flynn, and M. Guillemot. Browsing Recorded Meetings with Ferret. In *Proc. of MLMI 2004, Martigny, Switzerland*, pages 12–21, 2004.
15. K.-P. Yee, K. Swearingen, K. LI, and M. Hearst. Faceted Metadata for Image Search and Browsing. In *Proc. of CHI, Boston, USA, 2003*.

#	Tags	Utterance
738		maybe we can define it by like the demographic of
739		Right .
740		like h t how much money they have to spend or something like that ,
741	ID	Yeah . Yeah aim for a an income group .
742		like , well
743		That's a good point .
744		obviously it has to be someone who owns a television ,
745		Mm-hmm . Mm-hmm .
746		and like how recently have they bought their television like that sort of thing .
747		Yeah .
748		Mm .
749	?	So maybe it's more useful to d d to define objectives like fashion and simplicity than to find specific target group as far as age is
750		Yeah , t probably .
751	ID	Yeah .
752		because , yeah , things so different will appeal to different people ,
753		but
754		Okay . Um oh ,
755		there're a couple functions do we want something so that
756		do we want some kind of thing to find it if it's lost ?
757		Mm .
758		Yeah .

Figure 6. Proposed marker bar and line number for senetences in the transcript

PM	#423#	that I was just e-mailed right before the meeting
PM	#424#	and that we have to keep in in um in mind as we're creating this .
PM	? #426#	We didn't bring it up yet , or at all in the meetings so far ,
PM	? #427#	but we're not gonna work with teletext
PM	#428#	because um well it that's been taken over by the internet ,
PM	#429#	so we're not gonna worry about um we're not gonna worry about that .

Figure 7. Proposed Information Scent for the Summary Sentences