

A Visual Interface for Browsing and Summarizing Conversations

Shama Rashid *

Giuseppe Carenini †

Department of Computer Science
University of British Columbia

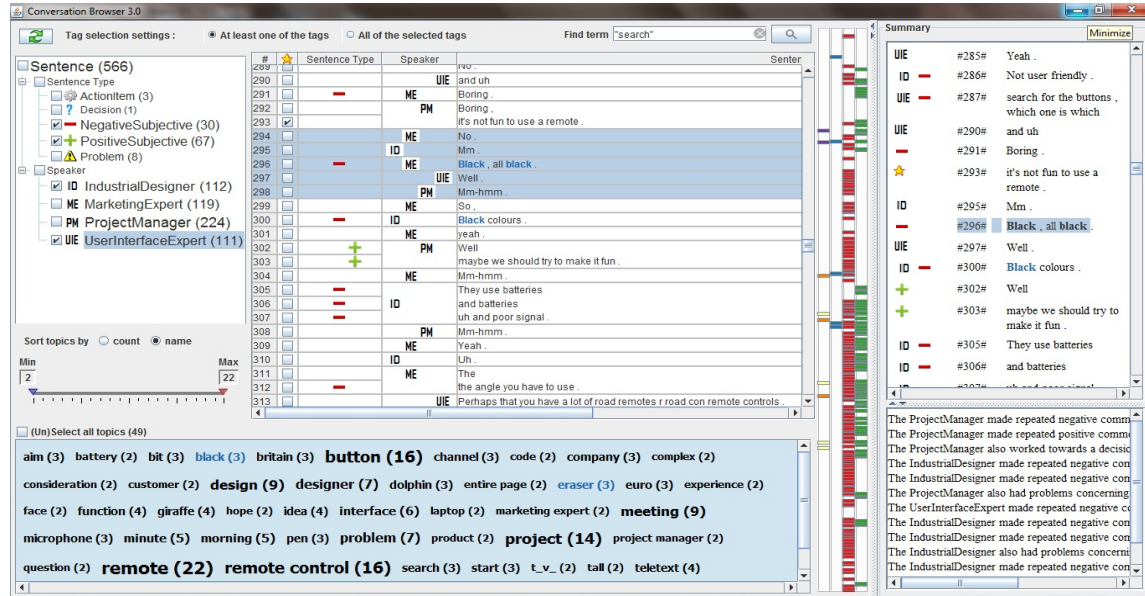


Figure 1: A visual interface for browsing and summarizing conversations with 4 integrated views - a) the Ontology View (left), b) the Transcript View (middle), c) the Entity View (bottom), and d) the Summary View (right)

1 INTRODUCTION AND RELATED WORK

In our daily lives, we have conversations with others in many different modalities like meetings, emails, chats, blogs etc. The Web has significantly increased the volume and the complexity of the conversational data generated through our day to day communication. Automatic summarization can be of benefit in dealing with this overwhelming amount of interactional information by providing quick access. However, due to its highly informal nature, the multimodal diverse data has proved to be a challenge from Natural Language Processing (NLP) point of view; rendering text processing using syntactic, lexicographic and semantic rules often insufficient. Here we present an interactive interface (see Fig. 1) that takes advantage of human perception in conjunction with automatically extracted conversation knowledge to generate a brief, focused overview of a conversation.

The dominant approach to summarization has been extractive, where the summary is generated by selecting and concatenating the most informative sentences from the source document(s). Extractive summarization has been popular as it is a simple binary classification task using machine learning techniques. The 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. The objective of our 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 the summary generated can be either extractive or abstractive focused summary.

Several projects in the past have investigated tools for browsing meetings. The Ferret Meeting Browser [9] 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. The Meeting Miner [2] aids browsing multimodal meetings through recordings of online text and speech collaborative meetings using timeline navigators of content of edits. 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 topics. For a more comprehensive list of relevant meeting browsers and document collection visualization tools please refer to [8]. Our interface differs from these other browsers because its purpose is to explore and summarize multi-modal conversations in general.

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 conversational data provided (offline) in a text format. However, for the time being we've been working mainly on the AMI meeting corpus [4].

2 DESIGN AND IMPLEMENTATION

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) type and En-

*e-mail: shama@cs.ubc.ca

†e-mail: carenini@cs.ubc.ca

tity [5]. 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), UserInterfaceExpert (UIE) and MarketingExpert (ME). The DA-type class, on the other hand, contains subclasses decision, action, problem, positive subjective and negative subjective sentences. The Entity class contains 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 including features like 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, our classifiers can be extended to multi-modal conversation data. We have also used sentence level features like word pairs, Part of Speech (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.’. This particular sentence has 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 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.

Our interface consists of four integrated views - the Ontology View, the Entity View, the Transcript View and the Summary View (see Fig. 1). The Ontology View provides a structured way for the users to explore the relevant concepts in the conversation and their relations. It contains a tree hierarchy with core nodes Speaker and DA type. The Entity View is a textual collage of the list of entities referred to in the conversation represented in a tag cloud format. Both of these views show a count beside every concept, within parenthesis, to indicate the frequency with which these concepts appear during the conversation. Also, the labels are scaled according to the counts to make the more frequent ones stand out more due to larger font. The Transcript View is designed to allow the users to inspect the whole conversation as well as the mapping of each sentence into the selected ontology concepts. The mapping is displayed by applying representative icons in the Sentence Type column for selected DA types and by highlighting the selected entities in bold blue font within the sentences themselves. The Participant icons are shown all the time in the Speaker column, once for each turn, to keep the user oriented to who is saying what; but selecting a Participant node on the Ontology View still acts as a filtering criteria. Whenever an ontology concept is selected, a marker appears corresponding to the relative position of its occurrence along the length of the conversation on a Marker Bar (one of the 4 vertical bars to the right of Transcript View). Different colored markers are used to indicate the concept type - blue for Entity, red for Participant, and green for DA type. The brighter shaded markers show the most recent selection of that particular concept type while the lighter shaded markers are for older selections. Marker Bars have been provided as a navigation mechanism on the interface. A user can click on a marker corresponding to a particular occurrence of a

selected concept (which particular concept is shown as tooltip text when the mouse pointer hovers over the marker) to jump to the part of the conversation where the corresponding sentence appears. We have also provided i) a generic Keyword Search Box, ii) a Tag Selection Settings that can be employed by the user to specify whether she wants to look for the sentences that refer to all of the selected tags or at least one of the selected tags, and iii) controls for the Entity View to shorten the list for easier handling.

The Summary View is a text area where the candidate summary of the conversation appears for user assessment. The summary is based on sentences selected from the transcript using the criteria set from the Ontology View and the Entity View. It provides an easier way to assess the conversation overview on a particular information need without scrolling through the whole transcript i.e. it acts as a filtered view of the sentences. The summary is also linked to the transcript and can act as a navigation tool by clicking on the summary sentences. The extractive summary sentences (displayed in the top panel of the Summary View) are prefixed with the tag icons to indicate why they were selected. The abstractive summary sentences (in the bottom panel) are followed by a list of the transcript sentences that were aggregated to generate it. We now briefly describe how the abstractive summaries are generated; more details can be found in [5, 6]. 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 an 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. An NLG component is used to generate a summary sentence corresponding to a message. The optimization function to sift more informative messages is $maximize(1 - \lambda) * \sum_i w_i s_i + \lambda * \sum_j u_j m_j$. 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.

3 DISCUSSION AND FUTURE DIRECTIONS

In the user study that we conducted to evaluate our interface, we found that different people take different approaches in navigating through a conversation and in trying to find informative sentences - some rely on the summary sentences (mainly extractive), some on basic keyword search, some on the markers etc. These results show the necessity for including all these components in our interface even though there is an overlap among their functionality.

Our current design has evolved through numerous iterations of small modifications applying HCI, NLP and InfoVis principles [3, 8]. Still, there is much scope for further improvement. We could reduce human cognition overload by dynamically adjusting the layout of the list of entities based on the keyword prefix the user types in the Search Box (placed on the top right corner of the Transcript View). In the future, we shall extend the interface to display non-linear asynchronous conversation threads like blogs and emails. This modification entails including a thread preview in the interface as in [7] and changes in the Transcript View. However, the classifiers for the DA types and the entity parsers are readily usable for other modes of conversational data. Recurring lexical 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 as in [1] may be helpful to the users to satisfy complex query needs. There is a number of ways the extracted entities could be clustered based on their semantic relationships or collocation; such a grouping will also reduce load on human cognition.

REFERENCES

- [1] 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.
- [2] M.-M. Bouamrane and S. Laz. Navigating Multimodal Meeting Recordings with the Meeting Miner. In *Proc. of FQAS, Milan, Italy, 2006*.
- [3] G. Carenini and G. Murray. Visual Structured Summaries of Human Conversations. In *Proc. of IVITA 2010, HongKong, China*, pages 41–44, 2010.
- [4] 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.
- [5] G. Murray and G. Carenini. Interpretation and Transformation for Abstracting Conversations. In *North American ACL, Los Angeles, CA, USA, 2010*.
- [6] G. Murray, G. Carenini, and R. Ng. Generating Abstracts of Meeting Conversations: A User Study. In *Proc. of INLG, Dublin, Ireland, 2010*.
- [7] 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.
- [8] 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*.
- [9] P. Wellner, M. Flynn, and M. Guillemot. Browsing Recorded Meetings with Ferret. In *Proc. of MLMI 2004, Martigny, Switzerland*, pages 12–21, 2004.