VISMEET: Data-Enhanced Video Conferencing

ANONYMOUS AUTHOR(S)

Recently, and especially as a result of COVID-19, video conferencing has become an integral part of our daily lives. However, modern video conferences are broadly viewed as poor substitutes for in-person meetings, plagued by technological issues and distracted participants. Here, we see an opportunity to use video conferences to improve in some aspects on the in-person meeting experience, much in the same way that televised sports can improve on certain aspects such as the cramped seats, bad weather, and poor sightlines associated with attending a real-life game.

In this paper, we present a novel system, VISMEET, for enhancing video conferences. It builds a dynamic and interactive transcript of the video conference and uses it to power a wide range of assistive visualizations and applications. For example, several visualizations help inattentive participants identify who has been speaking the most, what topics have been covered, and which meeting participants are likely expert on which topics. Other applications help with meeting time management and textual summarization of meeting contents.

We evaluated the effectiveness of conference-related visualizations in helping participants answer concrete questions about meeting content through a between-subjects study with 39 Amazon Mechanical Turk workers. The results show that when users watch the meeting video while viewing VISMEET visualizations, their accuracy in answering questions about meeting structure, topics, and speakers' contributions is 140% higher than when watching the meeting video alone. We also validate that the visualizations do not distract participants' attention from remembering detailed information available only in the meeting video.

CCS Concepts: • **Human-centered computing** → **Human computer interaction (HCI)**; **Interactive systems and tools**.

Additional Key Words and Phrases: Smart Meeting Rooms, Video Conference, Visualization

ACM Reference Format:

ANONYMOUS AUTHOR(s). 2021. VISMEET: Data-Enhanced Video Conferencing. In *CSCW '21, October 23–27, 2021.* ACM, New York, NY, USA, 16 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

Video conferencing today is broadly seen as at best a mediocre substitute for in-person meetings, with frequently degraded video and audio quality, rambling discussions, and environmental distractions. Even in meetings where conditions are ideal, some users tend to speak for too long, ideas and plans discussed during the meeting often get lost afterwards, and distracted meeting participants or participants who join late often lose context of the discussion and cannot catch up. While conferencing software will often produce meeting transcripts, spoken language is verbose and repetitive, and the resulting transcripts are tedious to navigate.

However, we view video conferences as an opportunity to *improve* the overall meeting experience and make it in some situations even better than the experience of in-person meetings through the considered application of technology. First, enriching transcripts by organizing them into section hierarchies and emphasizing key phrases and sentences can transform raw transcripts that are repetitive and time-consuming to read through into rich documents that are much easier to navigate. For example, participants who join late can quickly skim through the sections to get a sense for the topics that

Manuscript submitted to ACM

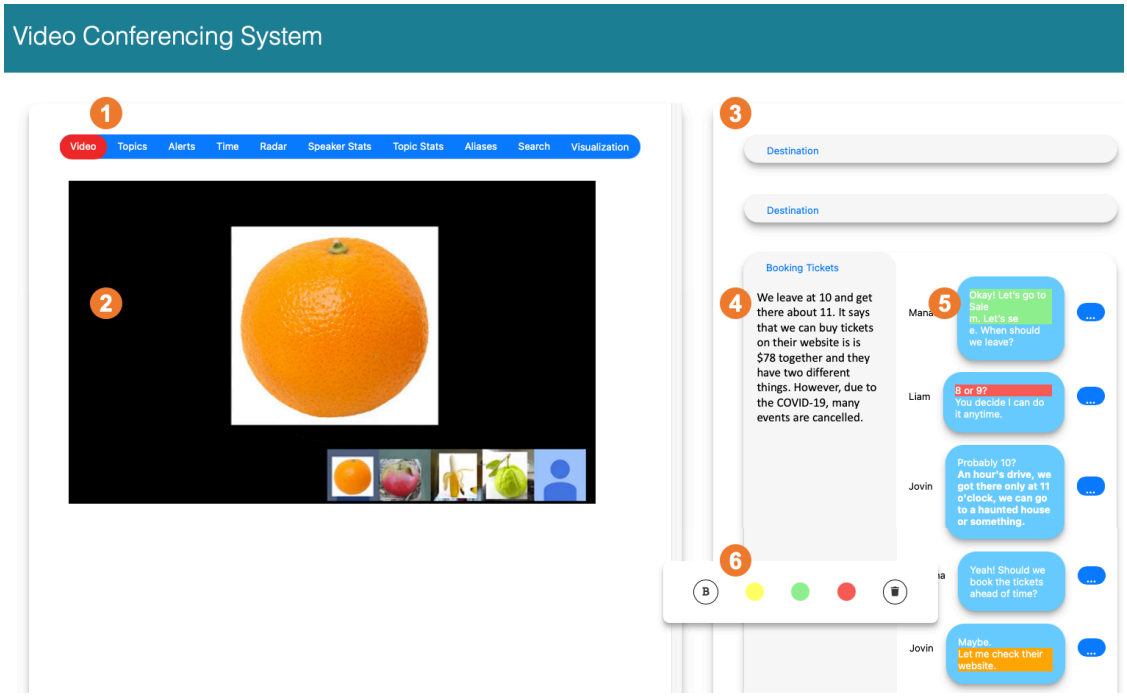

Fig. 1. VisMeet is a system for enhancing existing video conferences with additional information and applications derived from the transcript. Item (1) allows users to choose different information views. Item (2) shows the core video conference display screen, just as with most standard conference tools. Item (3) shows information extracted from the live meeting transcript. Item (4) is an automatically-generated summary of a section of the transcript. Item (5) shows automatically-highlighted sentences thought to be especially important. Item (6) shows that users can improve this feature by manually highlighting important text.

were already discussed earlier in the meeting. Similarly, if sentences corresponding to new ideas or next steps that were proposed during the meeting are extracted and integrated into a concise list, then forgetful participants could browse this list of ideas when deciding what to work on next, or use it to jump to the point in the conversation where a particular idea was discussed in detail.

Second, interactive user interface components can help participants better organize meetings and push discussion forwards. For example, a visualization that tracks speaker time can serve as a warning to participants to not dominate conversation. A visualization that covers speakers and topics can reveal which participants possess expertise in certain areas. A visualization of the topic discussion history can tell latecomers whether their pet topics have come up yet or not.

We developed VisMeet to leverage these opportunities: VisMeet captures the audio and video from a meeting, and processes this data to provide a visual user interface that enhances the video conferencing experience. The VisMeet interface sits alongside the gallery or speaker view provided by existing video conferencing software, and incorporates several applications built on top of a modular platform. The interface is dominated by an interactive, real-time transcript of the meeting. Applications provide functionality for users to organize and enhance this transcript in a semi-automated way. For example, the Meeting Topics application allows users to split the transcript into sections named based on the topic discussed during the section, thereby making the transcript much easier to navigate. Other applications provide real-time visualizations of meeting content, as found in the transcript.

To evaluate the effectiveness of VisMeet at improving awareness during video conferences, we conducted a between-subjects user study with 39 Amazon Mechanical Turk workers. Participants were asked to answer 13 questions related to a pre-recorded video of an online meeting. We found that VisMeet visualizations allowed participants to answer questions about meeting content dramatically more accurately than participants who had the videoconference content alone.

In summary, our contributions are:

- The VisMeet framework for instrumenting and enhancing video conferences.
- A video conferencing interface for offering users information and applications that improve the videoconference experience.
- A user study that shows our VisMeet visualizations can improve user accuracy when answering questions about meeting content by up to 140%.

2 RELATED WORK

2.1 Meeting Information Systems

The work that is closest to our own is probably the TalkTraces system by Chandrasegaran et al. [5] TalkTraces is designed to instrument and visualize the content of in-person meetings. The software presents a live and constantly-updated picture of a meeting's contents, via an in-meeting information display. Like our own system, TalkTraces makes heavy use of text derived from the meeting audio stream and uses it to produce information graphics that can aid meeting participants. The TalkTraces visualization tools are multifaceted and interrelated, designed to assist with overall meeting topic awareness. Evaluation of the TalkTraces system reported generally positive Likert-scale reactions from users.

Unlike TalkTraces, our system is designed solely for online video conferences. It is not clear that in-person and online meeting participants have similar needs. Our system aims to visualization different information content, often focused on comparatively simple reporting-style questions of who spoke on what topics when. Finally, our experimental evaluation does not rely on self-reported user feedback scores; rather it shows that users' factual recall of meeting content substantially improves when our tool is used.

Ehlen et al. [6] described a system for automatically extracting certain meeting-specific data artifacts — action items, topics, etc. — from the meeting transcript. Although some of their work presented a simple user interface for their data, the bulk of this paper focused on the mechanism for accurate extraction. In the future, VisMeet could potentially use such an extraction system as a component.

Tur et al. [16]'s CALO-MA system similarly aims to extract a meeting summary from the meeting audio stream. Their system has a particular focus on voice recognition methods, including extraction of dialog acts such as expression of a *command* or *agreement*. They did not present any information visualizations of the extracted data, nor any evidence about the impact of the extractions on users.

The SuVi system from Castronovo et al. [4] is, like much of our own system, designed to create visual summaries of meetings. However, SuVi focuses on building highly graphical storyboard- or newspaper-style layouts. The resulting meeting summaries are visually striking and compelling, but their usefulness is unclear, and it is not obvious how they could be used to inform meeting participants during the meeting itself.

2.2 Auto-Generated Transcripts in Video Conferences

Also relevant to our work are systems that apply automation and interactivity to meeting transcripts. SmartTranscript [8] proposes augmenting a transcript with automatically extracted topics and phrases. Users can select a topic to jump to

the messages in the transcript where it appears. Favre at al. [7] propose an efficient method for interactively correcting transcription errors through the use of clarification dialogs. In both of these works, development required the authors to build entire systems to handle audio capture of the meeting, processing audio to recognize speech, and applying natural language processing techniques. Thus, these works are examples of applications that could be explored much more efficiently with our VisMeet platform.

There has been considerable recent work in improving the accuracy of speech recognition. Deep Speech 2 [1] combines high performance computing methods with end-to-end deep learning to enable the rapid experimentation of new neural network models on vast natural language datasets. Zhang et al. [17] apply several layers of convolutional LSTMs with residual networks to improve recognition accuracy. Bahdanau et al. [2] develop an effective attention-based recurrent model for speech recognition by incorporating an n-gram language model. While these methods reduce transcription errors, they do not address the verbosity and repetitiveness of spoken language that makes transcripts difficult for users to leverage during and after meetings.

2.3 Immersive Video Conferencing Systems

In addition, there has been significant work on experiential, wholly immersive video conferencing systems which could potentially use our technology in the future, but as-is, these projects are not directly comparable to our work.

Several immersive video conferencing systems have been proposed to improve video conferencing by maximizing the information-richness communicated over the conference. Coliseum [7] aggregates video input from five cameras for each meeting participant to render a shared meeting space where participants can interact. Unlike conventional virtual worlds that use avatars, Coliseum renders meeting participants in the world based on their actual video. Blue-c [9] applies spatially immersive displays to effectively simulate a virtual surrounding window from one portal to another space. BiReality [12] uses a robotic surrogate to physically project a remote participant into a local room. The robot consists of a 360-degree display at head height that renders the participant.

Rather than focus solely on greater immersion, Hollen and Stornetta use the example of e-mail to argue that, instead, video conferencing systems should explore adaptions of face-to-face conversations that are appropriate for the Internet medium [10]. Several works have since pursued this idea. MirrorSpace [15] explores contrasting between blurry and sharp video based on the distance of a video booth to the closest person in deployments of pairs of booths. VideoProbe [11] explores the use of tablets installed in living spaces to share photos between family members living far apart. MyEyes [13] equips two users with virtual reality headsets, and overlaps their views to provide a shared experience. In Your Eyes [3] explores the use of video portals among couples living apart that can be called and answered without manual intervention, thereby approximating the potential for interruption in face-to-face interactions.

3 USER SCENARIO

Before introducing VisMeet, we first briefly introduce two scenarios that demonstrate how VisMeet can substantially improve the video conferencing experience.

Improved Meeting Understanding. Suppose Alex forgot to set an alarm last night and woke up late, thereby missing the first 20 minutes of a meeting using traditional video conferencing software such as Zoom. He listens for a few minutes, but isn't sure about what is currently being discussed — he skims over a real-time transcript that the software may provide, but it isn't very helpful because it lacks any organization. He ends up needing to interrupt the meeting to ask for a summary of what he's missed, causing delay for everyone else.

VisMeet creates a rich organized transcript by applying a semi-automated process, where annotations manually added by meeting participants are combined with ones automatically generated by an NLP toolkit specialized for transcript processing. The system also uses this information to generate customized live visualizations of the meeting's content so far. Using the information in these visualizations, Alex can tell that his area of specialty has not yet been discussed in the meeting; he can simply wait for his turn to speak, without any need for him to interrupt.

Speaking Time Breakdown. Suppose Samantha is participating with four other colleagues in a 30-minute meeting, with three topics on the agenda for discussion. However, Samantha inadvertently dominates the discussion of the first topic, and only 25 minutes into the meeting does she realize that the other participants have not had much of a chance to contribute to the discussion. Even worse, there is no longer enough time to discuss the other agenda items.

VisMeet incorporates applications that help participants quickly understand their relative speaking time compared to other participants on different topics during a meeting, as well as the progress of the meeting over time. Using VisMeet's Summary Diagram visualization, which incorporates a correlation diagram to summarize per-topic speaking times, Samantha could have more easily monitored and adapted her speaking. Similarly, she could have configured VisMeet's Meeting Topics applications to visually indicate when the meeting is running behind schedule; this automatically detects when the conversation has moved on to the next topic so that participants do not need to indicate it manually.

4 SYSTEM OVERVIEW

Our system, VisMeet, enhances video conferences with an interactive visual user interface that sits alongside the gallery or speaker view provided by existing video conferencing software. After starting a video conference on an existing video conferencing service (e.g., Zoom or Google Hangouts), a user can initialize a new instance of the system by providing the VisMeet system with the conference URL. The VisMeet system will join the conference as an additional robot participant, and it will populate the transcript in real-time by transcribing speech spoken in the conference.

The system then produces a web interface that renders the enhanced meeting information for all conference participants to access. We show the interface in Figure 1. The VisMeet framework provides a basic interface layout, but the UI can easily be customized for the user's particular needs. In this figure, our interface divides the transcript into sections displayed on the right, while additional features, including the video feed and various visualizations, are found in separate tabs on the left side of the page.

During the conference, participants can interact with the enabled VisMeet applications through the web interface. The system makes interactions collaborative by forwarding annotations added by one participant to all other participants, and it makes interactions persistent by storing annotations in a database. Following the meeting, the web interface remains active after the conference ends. Whereas the web interface operates as a live interactive tool during the conference, after the conference, it now functions effectively as a review tool to revisit the meeting later.

At the core of VisMeet is the real-time transcript of the meeting. By capturing the spoken content of the meeting, we use the transcript as a framework upon which we can build more useful applications and visualizations. In the section below, we further describe the enhancements we implemented.

5 SYSTEM APPLICATIONS

Additional applications built on top of the transcript provide a range of functionality, including time management and hierarchical summarization. We now detail the applications that VisMeet integrates into its system.

5.1 Important Sentences and Sentence Highlighting

As we have discussed, spoken language exhibits substantial redundancy, making it tedious to read raw transcribed speech. One way to improve the readability of a transcript is to emphasize sentences that summarize surrounding speech or that correspond to specific types of content, while de-emphasizing sentences that are either repetitive or not relevant to the discussion. For example, sentences that provide a high-level introduction to a new topic before the speaker begins discussing the topic in more detail may provide high-quality summaries; using visual effects to emphasize these sentences enables users to more efficiently skim through the transcript. Likewise, visually distinguishing various types of sentences, such as those corresponding to discussions of planned next steps, helps users efficiently locate specific content that they are interested in. On the other hand, meetings frequently include off-topic speech (e.g., "can you all see my screen?") that can be hidden to condense the transcript.

The important sentences application (⑤ in Figure 1) and sentence highlighting application (⑥ in Figure 1) implement sentence bolding, highlighting, and hiding to substantially enrich the transcript. First, the important sentences application allows users to mark a sentence as normal (default), important, or un-important. Sentences marked important appear bold in the transcript, while sequences of un-important sentences are collapsed to make it easier to scan through the transcript. On the other hand, the sentence highlighting application allows users to associate colors with particular content types (e.g., "future plans"), and to highlight selected sentences with those colors.

Both applications add buttons to the transcript interface to select and classify sentences. Additionally, both applications leverage the NLP toolkit so that, if the user so chooses, the system can learn to automatically reproduce the annotations made manually by users in previous meetings.

5.2 Meeting Topics and Section Hierarchy

Oftentimes, users looking back at a past meeting are interested in reading what was discussed about a specific topic. However, the flat format of a raw transcript makes it difficult to locate a particular discussion in the meeting. By breaking down the transcript into a series of topics and rendering the transcript as a hierachical document split into sections corresponding to those topics, VisMeet enables users to simply identify the section that they are interested in and search within that section.

The meeting topics application (③ in Figure 1) does exactly this. First, it adds a Topics tab to the interface where participants can plan out the topics that they intend to discuss during the meeting. Participants can use this topic list as an agenda for the meeting, and go through the topics in order; alternatively, it can be used just as a set of items that can be discussed in any order. The application also splits the transcript into sections, providing a button attached to each message that users can press to create a new section. When creating a new section, users can label the section with one of the topics defined ahead of time, or enter a new custom label. Adding a section hierarchy to the transcript greatly improves the organization of the transcribed speech, making it much easier to navigate the conversation.

The application also leverages the NLP toolkit to optionally provide some degree of automation. Although we explored applying the toolkit for automatic topic extraction, we found that this generally yielded low-quality topic outputs. Thus, we instead focus on automating the creation of sections by learning to associate sentences in the transcript with manually defined topics based on sections manually created by users in previous meetings. Then, while users still need to add topics at the beginning of a meeting, they do not need to provide further input during the meeting itself.

5.3 Section Summaries

Although splitting the transcript into sections labeled by concise topic names helps users navigate the transcript, it does not help participants who joined late or were distracted to get a quick understanding of what was discussed so far about each topic. Thus, the section summaries application (④ in Figure 1) adds a brief summary for each section that users can refer to for a quick overview of what was discussed about that topic, avoiding the need to read through all the transcribed speech in that section. When manually adding summaries, users are simply prompted to enter or edit a summary for the selected section.

Unlike in other applications, because state-of-the-art automatic text summarization techniques still generally produce low-quality summaries, we opt to employ a very different strategy in the automatic mode of this application — we do not learn from hand-written summaries in previous meetings, but instead build summaries by concatenating sentences that were classified as important by the important sentences application. When automatic section summaries are enabled, the summary simply consists of the three most "important" sentences in the section, i.e., the three sentences that have highest confidence score from the important sentences application.

5.4 Phrase-Based Notifications

Up to now, we have introduced applications that add functionality for enriching the transcript: these applications provide interfaces for manually inputting various annotations, and they can also be configured to use the NLP toolkit to learn to automatically produce these annotations. However, applications can also ingest the transcript for purposes that do not involve enriching the transcript. Phrase-based notifications is one such application, which neither adds new annotations to the transcript nor uses the NLP toolkit.

We developed this application because we found that, in some cases, a meeting participant is double-booked and needs to simultaneously attend multiple meetings, or only needs to be present in a long meeting at a particular short interval. Phrase-based notifications allows an individual user to enter keywords into a notification system, which will visually and audibly alert the user if any of those keywords are mentioned by other participants during the meeting. Thus, this helps users to ensure that they can participate when their presence is most needed.

5.5 Visualizations

Lastly, we develop several applications that focus on providing rich and concise visualizations of the meeting. We detail each of these visualizations below.

5.5.1 The Correlation Diagram of Meeting Topics and Speakers. In the Correlation Diagram of Meeting Topics and Speakers (① in Figure 2), each node on the left side represents a topic, and each node on the right side is a speaker. The height of each node reflects the time length spoken. For example, the height of the "Booking Tickets" node is short than the "COVID-19" node, which means speakers spend more time on the "COVID-19" topic. in Figure 2. The width of the flow between the two nodes represents the time that the speaker spends on the corresponding topic.

5.5.2 The Summary Diagram of Participants Performance. The Summary Diagram of Participants Performance (② in Figure 2) shows the speaking time each speaker spends across different meeting topics. The design of the circular bar chart makes the space occupied by the visualization consistent even if the number of participants is large. In each graph, each sector represents a speaker's speech time on the corresponding meeting topic. Different from the traditional circular bar chart, we set the scale interval to be equal to facilitate the display and comparison of shorter speaking times.

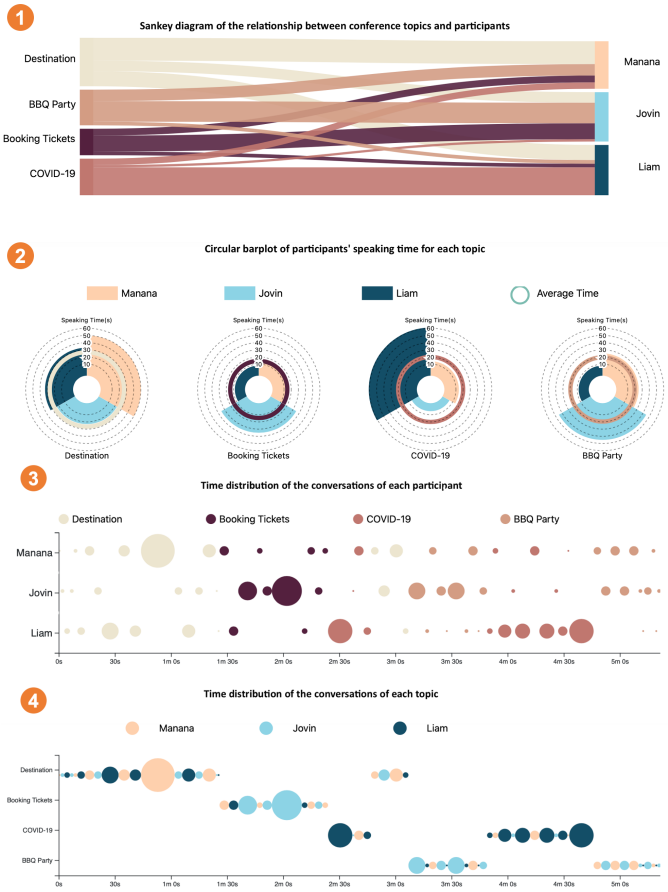

Fig. 2. The visualizations give users an overview of the meeting topics, speakers' performance, and important conversations from the video. (1) The Correlation Diagram of Meeting Topics and Speakers (2) The Summary Diagram of Participants Performance (3) The Distribution of Meeting Conversations Over Speakers (4) The Distribution of Meeting Conversations Over Meeting topics

In addition, we added a ring with the same color as the corresponding topic to show the average speaking time of that topic. A speaker can use this to determine whether to spend more time on a topic by comparing the height of the sector and the size of the ring.

5.5.3 The Distribution of Meeting Conversations Over Speakers and Meeting Topics. Several summative charts show the distributions of meeting conversations over speakers (③ in Figure 2) and meeting topics (④ in Figure 2). In both charts, we use grouped bubble charts to render the distributions of meeting conversations. The X-axis in both charts represents the timeline. For each circle, the radius indicates the time length of the corresponding conversation and the color represents either a topic or a speaker. In the distribution of meeting conversations over speakers, the y-axis shows speaker names. By clicking each circle, the corresponding transcript will be expanded on the message box. The y-axis in the distribution of meeting conversations over topics represents various topics. In this way, users can easily gain an overview of the order of topics and the occurrence of debates where multiple tiny circles with various colors appear.

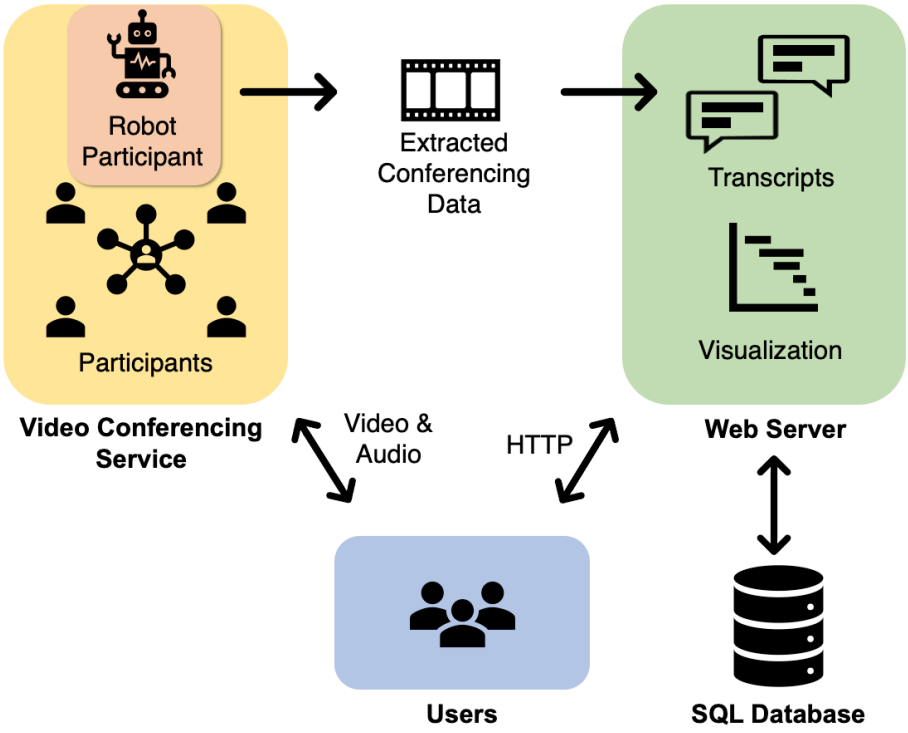

Fig. 3. The VisMeet system architecture.

6 IMPLEMENTATION

We summarize the system architecture in Figure 3. The robot participant and transcription service populate a VisMeet in real-time from a video conference, and a meeting manager and web interface maintain and render the meeting.

The VisMeet system consists of four components: the *robot participant* joins a video conference and extracts a stream of both the audio and the current speaker over time; the *transcription service* transcribes segments of audio corresponding to different speakers, yielding a stream of textual messages; the *meeting manager* runs the applications that have been enabled for a conference on the message stream, and serves HTTP endpoints; and the *web app* renders the transcript and enabled applications, and manages user interactions.

6.1 Robot Participant

The robot participant is a program which serves to provide a stream of audio and speaker names by joining the video conference as a virtual participant and extracting live data. Although the robot would not be necessary if we were to integrate VisMeet into a hosted video conferencing system such as Jitsi, we chose to provide a robot participant so that users can integrate our system using any service of their choice, including SaaS services such as Zoom and Google Hangouts.

Since the interface of video conferences varies widely between conferencing systems, the robot participant is implemented by a distinct, specialized plugin for each system. A general-purpose bot orchestrator manages a set of Docker containers, with one container for each ongoing conference, and the per-conference robot participants run inside these

containers. In our implementation, we only develop one participant plugin specialized for Google Hangouts, but this can be directly generalized to other meeting platforms as well.

Users register video calls with our system by providing the Hangouts meeting URL. Once a conference is registered, the bot orchestrator launches a new Docker container for the conference, and inside the container, the participant plugin (implemented as a Python program) opens a browser window, navigates to the specified URL, and enters the conference by pressing the Hangouts "join" button. The bot service destroys the container when the conference ends.

After joining the conference, the bot plugin extracts raw 16KHz audio by applying PulseAudio's parec program on the audio output monitor device. The bot emits the current speaker name at 10 Hz by injecting a small JavaScript program in the Hangouts webpage that repeatedly extracts the speaker name from the DOM; this name is transmitted to the Python bot plugin through a WebSocket connection. The plugin then passes the audio and speaker name streams to the bot service, which in turn forwards the streams to the transcription service.

The bot also records the conference to a video file by capturing the display in the container. This file is used to render the VisMeet interface after the conference ends, when the interface transitions to a review function.

6.2 Transcription Service

The transcription service processes the per-conference audio and speaker name streams into a single stream of transcribed messages. It performs transcription through a pluggable speech recognition provider, which theoretically could be performed locally (e.g. using Mozilla DeepSpeech[1]) or involve a SaaS API; in our implementation, we use the Google Speech-to-Text API[2]. The transcription service forwards the audio stream to the recognition provider, and the provider returns a stream of transcribed words that includes the timestamp in the audio stream when the word was spoken. The transcription service then concatenates the words spoken in a contiguous interval with no speaker name change into a message. The provider may also output end-of-utterance tokens, indicating a pause at the end of a sentence; the transcription service breaks up the words spoken by one speaker into one message per utterance so that, if one speaker talks for several minutes, messages are regularly emitted instead of only emitting one message after the speaker has changed.

6.3 Meeting Manager

The meeting manager maintains the per-conference message data, and runs the set of applications enabled for each conference on the message stream corresponding to that conference. We implement the manager in Go. The manager stores messages, sections, annotations, and application-specific data in an SQLite database. The manager serves both general-purpose and application-specific HTTP and WebSocket endpoints, such as listening for messages or annotations for a conference, or adding a new application-specific annotation.

6.4 Web Interface

The web interface renders the messages in the conference, as well as any application-specific user interface components; this is the core UI which a user would directly interact with. Our web interface is shown in Figure 1, where we display the meeting video and associated application components on the left, and we organize the enriched meeting transcript on the right. The applications, including our main visualizations as seen in Figure 2, are built directly onto the interface using JavaScript, and the site itself is built from a standard Bootstrap framework.

[1] Mozilla DeepSpeech is an open-source speech-to-text engine: https://github.com/mozilla/DeepSpeech.
[2] See https://cloud.google.com/speech-to-text.

6.5 Data Model

In order to store the details of the meeting, the meeting representation includes three components: the meeting transcript, a list of sections that group transcript messages into segments, and a set of application-specific annotations.

The transcript is an ordered sequence of messages, where a message is a list of contiguous sentences spoken by one meeting participant. Each message specifies the transcribed text, the speaker name, and the timestamp in the meeting. Sections specify an interval of messages that they group together, and include a label that describes the meeting activity during the section. Annotations can be created over entire sections, messages, or substrings of message text. An annotation specifies application metadata, e.g., a highlighted phrase may include the background color to use for highlighting.

6.6 Event Handling

To support the ease of development and integration with the application, the workflow of the framework is built atop a powerful event-handling infrastructure to allow for responsive changes.

Library events are processed serially by application code running on the web server. Message events are emitted when the transcript extractor creates a new message (i.e., when the speaker changes) or updates the text of an existing message (when more audio is processed through speech recognition).

Annotation events enable communication between applications, where one application uses the annotations of another. For example, an important sentences application may enable users to bold important sentences so that they are more prominent in the transcript. A section summary application may want to reuse these annotations to provide summaries for sections that are collapsed, by concatenating important sentences to construct summaries. Then, it can listen for "bold" annotation events from the important sentences application.

Applications may also listen for classification events on a particular annotation type and annotation granularity (section, message, or sentence). These events are used to automate interactions. When an application specifies a classification event listener, the system trains a classifier on the corresponding annotations made in prior meetings. Then, when the classifier has sufficiently high confidence for a new annotation, the library emits a classification event. The application can then decide whether to immediately create a new annotation based on the event, or to first get confirmation from the user; for example, an important sentence bolded by a user may immediately be visible to all participants, while an automatically bolded sentence may appear bold but have buttons to confirm or reject the automatic annotation. A classification event can be rejected, in which case the system uses this information to improve its performance.

To derive classification events, we train a logistic regression classifier on section, message, and sentence feature embeddings computed through Sentence-BERT [14], a variant of BERT optimized for sentence-level natural language processing tasks. Section and message embeddings are computed by averaging the embeddings of contained sentences. When a meeting initializes, the web server constructs a classifier for each type of classification event requested by applications. Then, when new sentences from the transcript extractor become available, we compute feature embeddings at different granularities and pass these embeddings to each classifier.

7 USER STUDY

We conducted a between-subjects study with thirty-nine participants to evaluate whether VisMeet could help them more effectively follow and understand the discussion in a video conference. Since our purpose is to test the effectiveness of providing the visualizations in assisting users to establish awareness of meeting data, we did not conduct this study in the entire VisMeet interface. Instead, we prepared a pre-recorded meeting video, which is used as the realistic

baseline condition. After filming a pre-recorded meeting video, we import it into our system to generate the visualizations (Figure 2).

7.1 Participants

We recruited 39 participants from Amazon Mechanical Turk (MTurk). Participants received $3 as compensation for their time. On the MTurk, we required that all participants should be qualified as Mechanical Turk Masters, which demonstrates the participants are excellent across a wide range of tasks and they could retain their high performance over time.

7.2 Protocol

For this study, we pre-record 5 minutes and 15 seconds online meeting video using Zoom with 3 meeting participants. In the online meeting, we carefully designed the conversations, which include 4 distinguished topics, Choosing a Travel Destination, Booking Tickets, COVID-19, and Barbecue Party respectively. We assigned one main speaker for each topic, which is the speaker with the longest speaking time. In the pre-recorded video, the main speaker of the topic Choosing a Travel Destination is Manana, the main speaker of Booking Tickets and Barbecue Party is Jovin, and the main speaker of COVID-19 is Liam. We selected three conditions and designed a corresponding questionnaire for each group.

- *Baseline*: Participants are only allowed to watch a pre-recorded video.
- VisMeet *Standard*: Participants are allowed to watch a pre-recorded video and four visualizations generated by VisMeet about this meeting. The pre-recorded video is shown on the same page with the visualizations so that participants can watch them at the same time.
- VisMeet *Without Video*: Participants are allowed to watch four visualizations generated by VisMeet about this meeting.

For each group, the differences are the materials we provided and the trap door questions on the questionnaire. For each group, participants were asked to answer 13 questions related to our pre-recorded meeting on the questionnaire. Each study took about 15 minutes. We inserted the video and visualizations before Section 0 in each corresponding questionnaire. When they were watching the video or visualizations, we allowed them to keep some notes, which is a common behavior in an online meeting. After watching the pre-recorded online meeting video and/or the visualizations, these materials are removed. Then they answered Section 0, which includes one or two trap door questions. Those questions are designed to check whether the participant has watched the video and/or the visualizations carefully. If they give the wrong answer to either question, their questionnaires will not be received.

Questions 1 to 11 are designed to evaluate the participants' awareness of the pre-recorded online meeting, and were all single selection questions. For each question, we also provided some necessary descriptions; for example, we added the description of "Total speaking time: 5:15" for question 1, "the main speaker is the speaker with the longest speaking time in each topic", etc. When the participants jump to the next section, the previous sections will be removed, and they cannot jump back to previous sections. Especially, for Question 12 and Question 13, we add separation between them so that their answer to Question 12 will not be influenced by the text of Question 13.

8 USER STUDY RESULTS

8.1 User Performance

After successfully receiving 39 questionnaires, we filtered the questionnaire based on their answers in Section 0. Seven of them did not successfully answer the trap door questions, which implies that the participants did not carefully watch the

Group	Score of Q1∼Q11		Precision Rate of Q12		Recall Rate of Q12		F1 Score of Q12		Score of Q13	
	Mean	Improvement	Mean	Improvement	Mean	Improvement	Mean	Improvement	Mean	Improvement
Standard Meeting Video (Baseline)	39.9%	-	0.58	-	0.23	-	0.41	-	38.46%	-
Visualization + Video	96.0%	140.35%	0.92	58.62%	0.63	173.91%	0.78	90.24%	84.62%	120.0%
Visualizations only	90.2%	126.32%	0.81	39.66%	0.33	43.48%	0.57	39.02%	30.77%	-20.0%

Table 1. User evaluation statistics for VISMEET.

video or visualizations. Among these 7 questionnaires, three of them belonged to the Baseline group, while two of them each belonged to the VISMEET Standard and VISMEET Without Video groups. We then discarded these 7 questionnaires and recruited 7 additional participants, confirming through the Worker ID provided by MTurk that they all did not fill in the previous questionnaires. After confirmation, we obtained 39 effective questionnaire responses in total, with 13 responses for each group. Careful inspection confirmed that there were no participants who answered multiple questionnaires or belonged to different groups at the same time.

We calculated the average score for Q1 to Q11. Since those who only watched the visualizations cannot answer Q12 and Q13, we only calculate the Precision, Recall rate, and F1 score for Q12, and we list the average score of Q13 in a single column. The Table 1 shows the results of our user study.

The study results reflect that providing visualizations significantly assist participants in building awareness of video conferencing data. Compared with the baseline group, both groups provided with visualizations gained higher scores in Q1 to Q11. This suggests that the groups provided with visualizations have a more comprehensive understanding of video conferencing structure, conferencing topics, and speakers' contribution, while the participants in the baseline group may not fully synthesize the implicit data within the online meeting video.

Questions 12 and 13 ask about the details mentioned in the conversations. Through these two questions, we evaluated whether providing the visualizations will distract the participants' attention from the actual meeting. The resulting precision and recall rates for Q12 in table 1 show that compared with only providing the meeting video, the participants provided the visualizations have higher average rates in both precision and recall. This result reflects that providing the related visualizations does not distract the participants' attention from the online meeting itself. In addition, this result even suggests that under certain circumstances, the participants may pay more attention to the details in the conversations when provided relevant visualizations. One possible reason is that when providing participants with visualizations, they gained some awareness of various meeting topics and the main speaker for each topic. They thus paid more attention to those conversations closely related to corresponding topics or spoken by the main speaker of that topic.

In Q13, the participants provided with both the video and visualizations see a huge improvement compared to the participants in the baseline group. This result further supports the remarkable contribution that providing visualizations in the online meeting can make in developing participants' awareness of video conferencing data. However, the participants provided with only visualizations performed worse than the baseline group. This result reminds us that even though providing the visualizations can make a significant difference in the general awareness understanding of meeting data, we will still need to observe the meeting itself to acquire more detailed information about the meeting.

We also found that providing the visualizations has a statistically significant impact in helping participants develop an awareness of conferencing data. On average, each participant who watched both the online meeting video and visualizations got a mean score of 96.0% (out of 100%) on Questions 1 to 11. In contrast, when the participants only watched the online meeting video, they got a mean of 39.9%. The mean difference between the Baseline (only given the online meeting video) and VISMEET Standard (given both the online meeting video and the visualizations related to it) groups over the scores of Questions 1 to 11 is statistically significant (paired t-test: $t\ \ 9.7857, \mathrm{df}\ \ 12, \text{p-value} < 0.00001$).

The comparison between the Baseline and VisMeet Standard groups indicates that providing the visualizations of the online meeting could significantly assist participants to build awareness of conferencing data, such as the meeting topics, speakers' contributions to each topic, and the progression of the conference as a whole. In addition, when only provided the visualizations of the online meeting, the participants attained an average score of 90.2%. The mean difference between the Baseline and VisMeet Without Video (only given the visualizations related to the online meeting) groups over their scores on Questions 1 to 11 is also statistically significant (paired t-test: t 9.1440, df 12, p-value $<$ 0.00001). Lastly, we found no statistically significant mean difference between the VisMeet Standard and VisMeet Without Video groups (paired t-test: t 1.4254, df 12, p-value 0.1795), which demonstrates that viewing the visualizations alone are sufficient in attaining an overarching understanding of the meeting's content, even if a user has not seen the video of the meeting itself.

The user study results suggest that providing the video conferencing participants with the visualizations related to the meeting data significantly aids in developing their awareness of meeting structure, topics, and speakers' contribution.

9 FUTURE WORK

Recurring Meetings. Although VisMeet's NLP toolkit is able to learn how to automatically organize transcripts based on previous meetings, VisMeet processes and summarizes each meeting independently. However, we find that often users are engaged in meetings that recur many times, e.g. weekly meetings over several months. In these scenarios, users often have difficulty identifying the specific meeting in which a particular topic was discussed, or understanding the evolution of an idea over several meetings. Thus, one important area for further work is in improved summarization and organization tools for recurring meetings.

Integrating Video from Multiple Cameras. Several users who tested VisMeet expressed dissatisfaction in how existing video conferencing technology handles multiple camera devices. For example, one professor had a multi-camera setup, with one camera directed towards their chair for normal usage, and other cameras pointed down at the desk, towards the whiteboard, etc.; the professor explained that they needed to manually change the camera input each time they shifted to a different position, leading to a cumbersome meeting experience. Thus, applying computer vision techniques to automatically identify the desired active camera, perhaps by learning from manual input changes in previous meetings if needed, or potentially even integrating video from multiple cameras into one video input, could make multi-camera setups substantially easier to use.

Visualizing More Types of Meeting Data. Currently, we can only visualize the data from the transcript extractor. There are more interesting and useful implicit data on the video conference, such as participants' emotions, gestures, and the identification of argument. By adding the emotion and gesture detection models to our system, we can analyze the status of participants in real-time. These features can help the main speaker better understand the real-time feedback from the participants and adjust the speaking speed. As for the existing visualizations, we want to present more quality-related measurements, such as average important words under each topic, or screen-sharing time. The results of Q12 and Q13 in our user study especially encourage us to extract more topic-related details. To achieve this, we would also need to implement a more advanced NLP model for transcript extraction.

10 CONCLUSION

In this paper, we have designed a platform that extracts the most important components of a video conference, and a suite of applications built on top of the platform that collectively form an effective visual user interface for improving the video conferencing experience. At the core, these components are the spoken messages of the meeting, along with information

on the speakers and timestamps of these messages. We conducted a between-subjects study with 39 participants on MTurk to evaluate the effectiveness of providing related visualizations on the online conferencing system. The study results show that when providing related visualizations, participants gained significantly better scores in understanding the topics, speakers' contributions, and the overall process of the meeting. We also found that the visualizations can slightly improve the participants' attention to remembering detailed information in the online meeting conversations.

REFERENCES

[1] Dario Amodei, Sundaram Ananthanarayanan, Rishita Anubhai, Jingliang Bai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro, Qiang Cheng, Guoliang Chen, et al. 2016. Deep speech 2: End-to-end speech recognition in english and mandarin. In *International conference on machine learning*. PMLR, 173–182.

[2] Dzmitry Bahdanau, Jan Chorowski, Dmitriy Serdyuk, Philemon Brakel, and Yoshua Bengio. 2016. End-to-end attention-based large vocabulary speech recognition. In *2016 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 4945–4949.

[3] Uddipana Baishya and Carman Neustaedter. 2017. In your eyes: Anytime, anywhere video and audio streaming for couples. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*. 84–97.

[4] Sandro Castronovo, Jochen Frey, and Peter Poller. 2008. A generic layout-tool for summaries of meetings in a constraint-based approach. In *International Workshop on Machine Learning for Multimodal Interaction*. Springer, 248–259.

[5] Senthil Chandrasegaran, Chris Bryan, Hidekazu Shidara, Tung-Yen Chuang, and Kwan-Liu Ma. 2019. TalkTraces: Real-time capture and visualization of verbal content in meetings. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–14.

[6] Patrick Ehlen, Matthew Purver, John Niekrasz, Kari Lee, and Stanley Peters. 2008. Meeting adjourned: off-line learning interfaces for automatic meeting understanding. In *Proceedings of the 13th international conference on Intelligent user interfaces*. 276–284.

[7] B. Favre, M. Rouvier, and F. Bechet. 2014. Reranked aligners for interactive transcript correction. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 146–150. https://doi.org/10.1109/ICASSP.2014.6853575

[8] Cornelius Glackin, Nazim Dugan, Nigel Cannings, and Julie Wall. 2019. Smart Transcription. In *Proceedings of the 31st European Conference on Cognitive Ergonomics*. 134–137.

[9] Markus Gross, Stephan Würmlin, Martin Naef, Edouard Lamboray, Christian Spagno, Andreas Kunz, Esther Koller-Meier, Tomas Svoboda, Luc Van Gool, Silke Lang, et al. 2003. blue-c: a spatially immersive display and 3d video portal for telepresence. *ACM Transactions on Graphics (TOG)* 22, 3 (2003), 819–827.

[10] Jim Hollan and Scott Stornetta. 1992. Beyond being there. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 119–125.

[11] Hilary Hutchinson, Wendy Mackay, Bo Westerlund, Benjamin B Bederson, Allison Druin, Catherine Plaisant, Michel Beaudouin-Lafon, Stéphane Conversy, Helen Evans, Heiko Hansen, et al. 2003. Technology probes: inspiring design for and with families. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 17–24.

[12] Norman P Jouppi, Subu Iyer, Stan Thomas, and April Slayden. 2004. Bireality: mutually-immersive telepresence. In *Proceedings of the 12th annual ACM international conference on Multimedia*. 860–867.

[13] Rui Pan, Samarth Singhal, Bernhard E Riecke, Emily Cramer, and Carman Neustaedter. 2017. " MyEyes" The Design and Evaluation of First Person View Video Streaming for Long-Distance Couples. In *Proceedings of the 2017 Conference on Designing Interactive Systems*. 135–146.

[14] Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084* (2019).

[15] Nicolas Roussel, Helen Evans, and Heiko Hansen. 2004. Proximity as an interface for video communication. *IEEE MultiMedia* 11, 3 (2004), 12–16.

[16] Gokhan Tur, Andreas Stolcke, Lynn Voss, Stanley Peters, Dilek Hakkani-Tur, John Dowding, Benoit Favre, Raquel Fernández, Matthew Frampton, Mike Frandsen, et al. 2010. The CALO meeting assistant system. *IEEE Transactions on Audio, Speech, and Language Processing* 18, 6 (2010), 1601–1611.

[17] Yu Zhang, William Chan, and Navdeep Jaitly. 2017. Very deep convolutional networks for end-to-end speech recognition. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 4845–4849.

A USER STUDY QUESTIONNAIRE

- *Section 0 (Trap Door Question)*:
  - *(Only for Baseline and* VisMeet *Standard)*What is the special word mentioned on the video?
  - How many speakers are there in the meeting?
- *Section 1 (Talking time of participants)*:
  - *Q1* Please estimate which of the following is closest to the speaking time of speaker Jovin.

  - *Q2* Please estimate which of the following is closest to the speaking time of speaker Liam.
  - *Q3* Please estimate which of the following is closest to the speaking time of speaker Manana.
- *Section 2 (Arrange the topics of the meeting in the order of occurrence)*:
  - *Q4* Which topic was the 1st mentioned?
  - *Q5* Which topic was the 2nd mentioned?
  - *Q6* Which topic was the 3rd mentioned?
  - *Q7* Which topic was the 4th mentioned?
- *Section 3 (Find the main speaker for each meeting topic)*:
  - *Q8* Who is the main speaker of the topic: "Choosing a Travel Destination"?
  - *Q9* Who is the main speaker of the topic: "Booking Tickets"?
  - *Q10* Who is the main speaker of the topic: "COVID-19"?
  - *Q11* Who is the main speaker of the topic: "Barbecue Party"?
- *Section 4 (Conversation Details)*:
  - *Q12* When we discussed destinations, which locations were mentioned?
  - *Q13* Why did the participants decide not to go to Salem?