RTDS: Real-Time Discussion Statistics

Pascal Bissig, Jan Deriu, Klaus-Tycho Foerster, Roger Wattenhofer
Distributed Computing Group, ETH Zurich
Gloriastrasse 35, CH-8092 Zurich, Switzerland
{firstname.lastname}@tik.ee.ethz.ch

ABSTRACT

We present *RTDS*, an Android application to analyze discussions while they are taking place. Using two microphones of a smart phone and Time Difference of Arrival measurements, conversations of participants are evaluated regarding, e.g., speaking time, contributions, or complex interaction patterns. The application can also assume the role of an active referee to ensure that all speakers get a fair share of the ongoing conversation. By using an off the shelf smart phone with two microphones, our system can immediately be applied to track spoken interactions between people. Experimental results show that our implementation causes only 2% user classification errors whilst being able to run in real-time on a standard smart phone without hardware modifications.

Author Keywords

Vocal interaction; discussion analysis; Angle of Arrival measurement; Markov modeling; mobile devices.

ACM Classification Keywords

H.5.2 Group and Organization Interfaces: Voice I/O

INTRODUCTION

Modern technology has allowed us to quantify more and more aspects of our behavior. A few years ago, fitness tracking for example was limited to comparing lap times, running distances or other key parameters of a workout. Today, fitness tracking devices can tell you, with high accuracy, how much exercise you get throughout a normal day by continuously monitoring your behavior. The ease of quantifying complex personal exercise behavior leaves a user with undeniable facts, that when taken seriously, can help improving quality of life.

The same principles have been applied to social interactions. For example, there is a tool that helps quantifying the quality of the encounters with your friends [13].

However, current tracking methods for interactions are far from being convenient to use. To our knowledge there are no tools to easily track verbal interactions, even though conversations are an integral part of our everyday lives and the underlying mechanics might give great insights on the state of a relationship. Maybe you have the feeling that you are always interrupted by a specific person during work meetings, but want to quantify that feeling into numeric values? With *RTDS* we present a solution (not only) for this issue.

RTDS can accurately capture how people interact in a discussion. This tool can give insights about how much people speak, or reveal more complex patterns in their interaction. *RTDS* is applicable in many situations such as business meetings, interviews, dinner conversations, or during arguments. Running on of the shelf smartphones, *RTDS* is quickly setup and can give insights about verbal interaction patterns wherever you go. Furthermore, *RTDS* can also augment conversations by acting as a referee, e.g., by notifying participants of using too much time in the ongoing conversation.

Contribution and Ongoing Work

Our contribution in this short paper focuses on showing the technical feasibility of the described system on an off the shelf smart phone, combined with a pilot evaluation.

For a future full version of this paper, we plan to incorporate multiple extensions discussed at the end of this work, developed jointly with user studies – going beyond the technical contribution of our short paper.

RELATED WORK

Our discussion mechanics inference method is based on *Time Difference of Arrival (TDOA)* to determine the *Angle of Arrival (AOA)* of sound signals. Humans and animals use this method in everyday life with their two ears to aid the localization of sounds, we refer to [11] for an overview: As it turns out, the change of the angle of a sound source can be detected by humans even if the difference is as small as 1°, depending on the relative position of the sound source. Not surprisingly, this established technique has already been applied in various contexts, e.g., shooter detection: In Washington, D.C., USA, the installment of just 300 sensors was enough to localize several ten thousands of gun shots in an area of 20 square miles since 2006 [10].

Another type of work is using TDOA to localize the (single) device itself. Global navigation satellite systems, such as *GPS*, *Galileo*, or *GLONASS*, employ TDOA correlation to determine, e.g., the position of a smart phone [5]. However, they use radio waves instead of acoustics and have further information about the senders.

MUM '16, December 12-15, 2016, Rovaniemi, Finland

DOI: http://dx.doi.org/10.1145/3012709.3012726

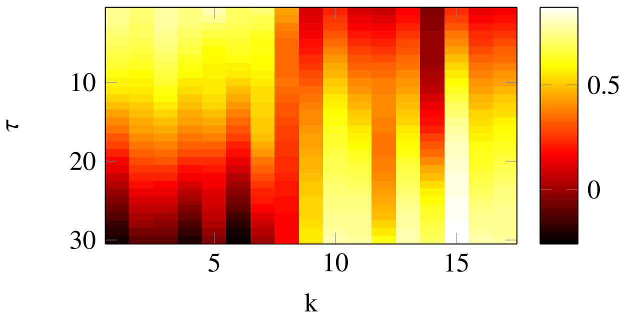


Figure 1: Values of $c(\tau)$ for different time slices k (44100 samples each) and sensible values of τ considering a sampling rate of 44.1kHz and microphone distance of 12.7 cm. At $k = 8$, the TDOA changed drastically which is clearly reflected in the change of $c(\tau)$.

A different line of research leveraging TDOA focuses on *around device interaction*: E.g., *Toffee* [16] uses an acoustic TDOA approach to detect the location of nearby (finger) taps. Devices, such as smart phones or laptops, are augmented with four vibro-acoustic sensors, located below the device. Then, the devices, which are placed on a hard surface, can use the relative angle of the finger taps as an input method, e.g., to switch desktops or to control a game. They note that "*distance estimation is too poor for practical use at this time*". Besides a distinct application, a main difference is that they use customized hardware, whereas we do not.

MEASUREMENT SYSTEM

Application wise, more linked to this paper is using TDOA to determine the relative speaker location/angle (cf. [2, 4]) or to localize a set of connected smart phones in a meeting, cf. [9], where 10 phones are used for accuracy. However, all these systems differ from our work in the sense that we just need one off the shelf smart phone with our application installed. No specialized hardware, multiple phones, or long setup-times are needed. The participants enroll with one tap on the screen.

McCowan et al. [8] presented a system that automatically analyzes and stores meeting contents. Amongst others, features such as "speech pitch" and "presentation speech activity" are tracked. The features allow for fine grained analysis of the meeting. However, the system requires multiple cameras and microphones being placed in a certain way. This reduces the applicability in many real world scenarios in which *RTDS* can give insights on the social dynamics of a discussion.

Another aspect of our work relates to *lifelogging* (cf. [1, 3]) & *the quantified self* (e.g., [14]) and augmenting conversations with smart phones. Many people are interested in quantifying activities in their everyday life by logging (for example, recording audio with their smart phone all the time [12]) and subsequently analyzing them. Automatically augmenting discussions can come into play here, by, e.g., trying to enforce certain conversation criteria (as done in our work). Current applications such as the system in [6] embark in a similar (yet maybe orthogonal) direction by creating, e.g., tickets-to-talk between participants.

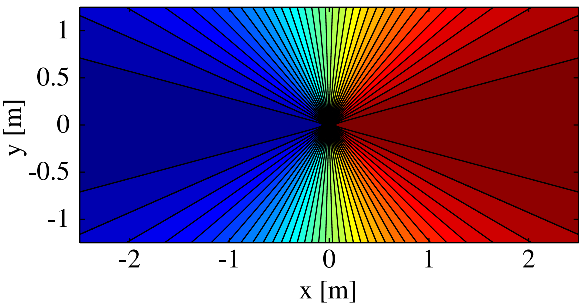


Figure 2: Areas of same color lead to the same TDOA measurement at the phone. The area shown is 5m by 2.5m and the microphone assumptions are based on the Samsung I9300 which we used for our experiments. As seen, TDOA with two microphones cannot distinguish two speakers sitting on opposite sides of the microphone axis.

In order to distinguish people within one discussion, we rely on a Time Difference of Arrival measurement of the audio signal. We use a Samsung I9300 smart phone from 2012 which allows for stereo recording using the two built in microphones. A large portion of current smart phones is equipped with two microphones, with some even having three or more (e.g., LG G3). The microphones are sampled at 44.1 kHz and are located at the top and bottom of the device (12.7 cm apart). Assuming the spoken signal is $s(t)$, we assume that we receive $r_i(t) = k_i \cdot s(t-\tau_i)+n(t)$ at microphone i. The noise term $n(t)$ is assumed to be uncorrelated and the signal $s(t)$ is shifted in time due to the distance between microphone and the sound source. The anisotropic gain of the microphones and the speaker itself is captured in k_i as it is constant during a discussion as long as we assume neither microphones or participants move significantly. To find the angle of arrival based on two microphones ($i \in [1, 2]$), we use a cross correlation of the received signals: $c(\tau) = \sum_{i=0}^{n} r_1(i) \cdot r_2(i - \tau)$.

This gives us an indication of how well the received signals match for a given time delay τ within a fixed time slice of length n samples. The value τ which maximizes $c(\tau)$ gives the time difference of arrival at which the most energy arrives at the two microphones. Since each τ corresponds to an angle of arrival given the distance between the microphones (shown in Figure 2, the speakers can approximately be localized with respect to the microphone pair. Figure 1 shows values of $c(\tau)$ for different delays τ over a period of 17 consecutive time slices of one second. The change of $c(\tau)$ at half time of the measurement indicates that the angle of arrival has changed significantly. See Figure 2 to get an intuitive idea of which areas around the microphone pair lead to a fixed value of τ. As you can see, the areas extend to both sides of the microphone axis. Therefore, two speakers sitting exactly opposite of each other with respect to the microphone axis cannot be distinguished using our method and only two microphones. However, this problem can easily be avoided by placing the

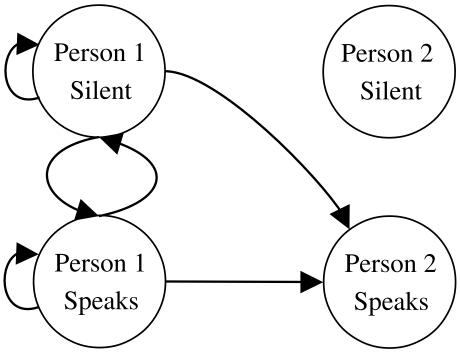


Figure 3: Hidden Markov Model template for two speakers. Silent states are introduced for each user to capture user specific discussion behavior. For simplicity, only transitions originating from Person 1 are drawn.

phone such that the arrival angles are unambiguous. Should there be no dominant angle of arrival ($c(\tau) < x \; \forall \; \tau$), we assume that nobody was speaking. The threshold we use to separate silence from speech is empirically evaluated (we refer to the evaluation section for details).

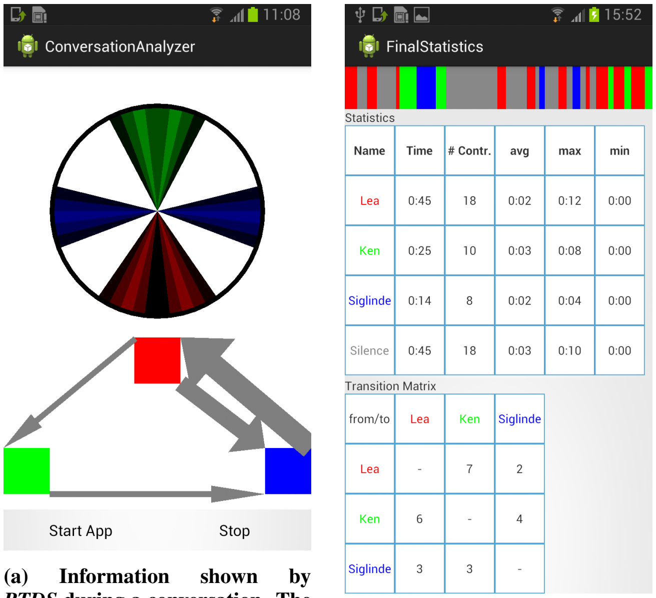
Discussion Mechanics Inference

In order to reduce the effect of measurement noise and to infer high level discussion statistics, we use a Markov model. The structure of the model is defined as shown in Figure 3. Each user can either be speaking or silent (top and bottom row of states). By modeling silent states for each user, we allow the model to distinguish a pause from a user continuing the discussion after another stopped talking. The transition probabilities are adapted throughout each discussion to best explain the sequence of angle of arrival measurements using the Baum-Welch algorithm, cf. [7]. The emission probability distributions for each speaking state matches the dimensions of the TDOA measurements. Figure 6 shows such a distribution that resulted from one of our test discussions. Each silent state is only allowed to emit silence observations. This means that our system does not require prior information about where speakers are located, and is able to find and distinguish speakers solely based on the structure of the Hidden Markov Model. To generate the final statistics, we use the Viterbi algorithm [15] to obtain the most likely state sequence using the trained model.

APPLICATION

We implemented the discussion inference method described above in an Android application. The application requires two microphones that can be programmatically accessed which most modern phones provide. In order to obtain statistics for a discussion, each participant needs to specify an alias and an approximate direction so the states of the Markov chain can be matched to the user aliases.

During any discussion, *RTDS* provides on-line feedback about the current state of the discussion as shown in Figure 4a.



Statistics

Name	Time	# Contr.	avg	max	min
Lea	0:45	18	0:02	0:12	0:00
Ken	0:25	10	0:03	0:08	0:00
Siglinde	0:14	8	0:02	0:04	0:00
Silence	0:45	18	0:03	0:10	0:00

Transition Matrix

from/to	Lea	Ken	Siglinde
Lea	-	7	2
Ken	6	-	4
Siglinde	3	3	-

(a) Information shown by *RTDS* during a conversation. The circle shows the arrival angles for each user in a different color. The symmetry of these arrival angles is caused by the fact, that two microphones only allow for unambiguous angle arrival detection to either side of the microphone axis. Below, the transition probabilities between all the speakers in the discussion are shown. Wider arrows indicate higher transition probabilities.

(b) An example for the final statistics shown by *RTDS*. Included is the speaking time for each participant (Time) and the number of distinct contributions (# Contr.). In the lower part of the screen the complete transition matrix between the users is shown (silent and speaking states for each user are fused). The bar at the top displays who spoke during which time period.

Figure 4: Two of the multiple (display) modes of *RTDS*.

Offline Feedback

In addition to the online feedback, *RTDS* provides detailed statistics after each discussion, cf. Subfigure 4b. These include the total amount of time each participant spoke and how many distinct contributions were made. Also the min, max and average length of each contribution is listed for each participant. The transition matrix shows how many contributions from participant A were followed by a contribution of participant B. This information can be useful for finding repeating patterns in the way the participants interact.

Special Operation Modes

In addition to the passive feedback modes described above, we can use the online statistical data to enforce constraints on how participants may interact. In case a constraint is violated, we can give visual or acoustic feedback. We included two operating modes into *RTDS* that the user can choose from. The first mode helps enforce even speaking times for all participants. The second mode forces participants to take turns when talking and limiting the talking time for each participant and round. These modes for example can be used to defuse a heated argument by disallowing repeated interruptions or by reproving participants that are not allowing others to talk.

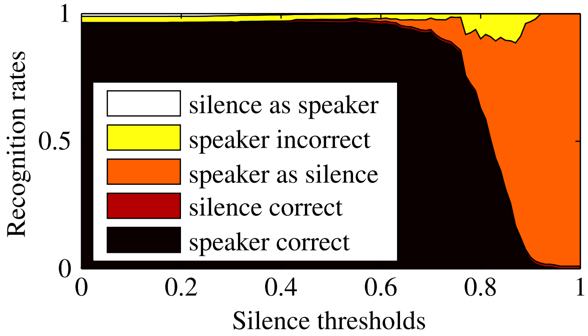


Figure 5: The graph in this figure depicts how the silence threshold (in $\frac{\max(c(\tau))}{\text{sum}(c(\tau))}$) affects the classification accuracies of both speaking and silent states.

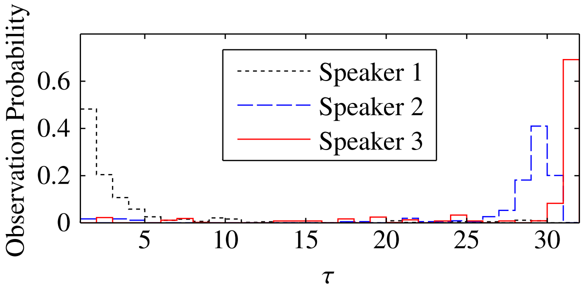


Figure 6: The observation probability distributions are accurately estimated even if two users cause similar time of arrival differences τ depicted on the x-axis. In this case, speaker 1 and speaker 2 are only separated by a two sample delay between the microphone channels.

PILOT EVALUATION

To evaluate the accuracy of our system, we recorded 6 separate discussions with a total of 7 people. Each of the 6 discussions was annotated manually, this information was then used as ground truth. In all 6 discussions, two (1), three (3), or four (2) people were involved, and the phone was placed in a way that allowed for unambiguous arrival angles for all participants. Natural disturbances like coffee machines or other background noise were present. Figure 5 shows the overall classification results with respect to a varying silence threshold. We consider direct transitions between speaking states of two users a valuable metric that captures how one user might disrupt another. Therefore, we consider confusing speakers worse than confusing a speaking state with silence since the latter does not affect direct transition probabilities between speaking states. With a growing number of observations being classified as silence, more speaking states are confused with silence. Most wrong speaking user classifications occur at a silence threshold of 0.8-0.9. For lower threshold, the model matches the discussion better thereby causing less errors. For higher thresholds, too many speaking states are mistaken for silence, thereby reducing the share of speaker misclassification. According to our results, a silence threshold of 0.5 is ideal as it leads to error rates of 3% and erroneous speaking user classification rates of only 2%.

All our experiments were carried out with up to four participants. However, the number of speakers is not limited to four. Figure 6 shows the observation probabilities in a discussion during which two speakers (2 and 3) sat very close to each other and therefore caused similar time difference of arrival measurements. Even though the measurements are only 2 samples apart on average ($45\mu s$), speakers are reliably detected and misclassification rates are only 2%.

FUTURE WORK

For future versions of *RTDS*, we envision multiple extensions, such as *i*) more in-depth statistics, *ii*) enhanced referee functions, *iii*) usage of multiple smart phones, and *iv*) integrated voice recognition to deal with moving participants. For an early prototype version of *RTDS*, we experimented with voice recognition to identify participants: While the recognition rates were in general still acceptable (mostly close to Angle-of-Arrival), there were accuracy issues with recognizing very similar voices, e.g., brothers.

CONCLUSION

In this paper we presented *RTDS*, a smart phone application to generate participation & interaction statistics about conversations. Our application works online and offline, displaying useful statistics during the conversation and summarized information afterwards. Furthermore, *RTDS* can be used as an active referee to enhance discussions by, e.g., allocating a fair share of time to each participant. We did not use any specialized hardware and reach a high accuracy with an Angle-of-Arrival measurement and a hidden Markov model.

REFERENCES

1. Megan Bearder and Stephen Wolfram. 2013. Stephen Wolfram on Personal Analytics. MIT Technology Review. (May 2013).

2. Michael S. Brandstein and Harvey F. Silverman. 1997. A practical methodology for speech source localization with microphone arrays. *Computer Speech & Language* 11, 2 (1997), 91 – 126.

3. Vannevar Bush. 1996. As We May Think, (Reprint from 1945). *interactions* 3, 2 (March 1996), 35–46.

4. Simon Haykin and K.J. Ray Liu. 2010. *Handbook on Array Processing and Sensor Networks*. Wiley-IEEE.

5. Bernhard Hofmann-Wellenhof, Herbert Lichtenegger, and Elmar Wasle. 2008. *GNSS: GPS, GLONASS, Galileo and more*. Springer.

6. Pradthana Jarusriboonchai, Thomas Olsson, and Kaisa Väänänen-Vainio-Mattila. 2014. User Experience of Proactive Audio-based Social Devices: A Wizard-of-oz Study. In *MUM'14*.

7. Christopher D. Manning and Hinrich Schütze. 1999. *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, MA, USA.

8. Iain McCowan, Daniel Gatica-Perez, Samy Bengio, Guillaume Lathoud, Mark Barnard, and Dong Zhang. 2005. Automatic analysis of multimodal group actions in meetings. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 27, 3 (2005), 305–317.

9. M. Parviainen, P. Pertila, and M.S. Hamalainen. 2014. Self-localization of wireless acoustic sensors in meeting rooms. In *HSCMA '14*.

10. Andras Petho, David S. Fallis, and Dan Keating. 2013. ShotSpotter detection system documents 39,000 shooting incidents in the District. The Washington Post. (November 2013). `http://wapo.st/16vVyx1`.

11. Jan Schnupp, Israel Nelken, and Andrew King. 2011. *Auditory Neuroscience : Making Sense of Sound*. Cambridge, Mass. MIT Press.

12. M. Shah, B. Mears, C. Chakrabarti, and A. Spanias. 2012. Lifelogging: Archival and retrieval of continuously recorded audio using wearable devices. In *ESPA '12*.

13. Liz Stinson. 2015. Having a Hard Time Being a Human? This App Manages Friendships for You. WIRED. (January 2015). `http://www.wired.com/2015/01/hard-time-human-app-manages-friendships/`.

14. Melanie Swan. 2012. Sensor Mania! The Internet of Things, Wearable Computing, Objective Metrics, and the Quantified Self 2.0. *Journal of Sensor and Actuator Networks* 1, 3 (2012), 217–253.

15. A.J. Viterbi. 1967. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *Information Theory, IEEE Transactions on* 13, 2 (April 1967), 260–269.

16. Robert Xiao, Greg Lew, James Marsanico, Divya Hariharan, Scott Hudson, and Chris Harrison. 2014. Toffee: Enabling Ad Hoc, Around-device Interaction with Acoustic Time-of-arrival Correlation. In *MobileHCI '14*.