

ScoringTalk: 発話量の可視化と採点に基づき 発話機会を調整するタブレットシステム

安達 寛之^{1,a)} 明神 聖子^{1,b)} 島田 伸敬^{1,c)}

概要: 一般に、複数人の会話では、頻繁に発話する人とあまり発話しない人がおり、発話の少ない人から十分な情報が得られないことがある。また、話す人がより話し、会話が一方通行になりがちなことがある。そこで本研究では、そのような会話をコントロールし、聞き手になりがちの人に話すチャンスを与え、発話の多い人には他者との発話のバランスを考慮しながら会話してもらえよう、ゲーミングフレームワークを考えている。そのための試みとして、複数人の会話において、ユーザの発話加減(どのくらいの時間、誰に話しかけているか)をリアルタイムに可視化するとともに、採点をしてユーザにフィードバックすることで、ユーザが発話機会を調整できるシステムを開発する。本稿では、そのシステムの構成と機能について説明し、システムの有無による発話量の均衡化の程度について比較実験を行い、その結果を議論する。

ScoringTalk: Tablet System that Controls Timing of Utterance by Visualizing and Scoring How Long Users Speak to Whom

HIROYUKI ADACHI^{1,a)} SEIKO MYOJIN^{1,b)} NOBUTAKA SHIMADA^{1,c)}

Abstract: In general, someone speaks more often or less than others in multi-party conversation. We cannot obtain enough information from those who speak less than others. Moreover, sometimes one speaks too much, and it tends to be a one-way conversation. Therefore, we are thinking a gaming framework to break through such a conversation, which makes someone only listening have a chance to speak and makes someone speaking more often than others to consider the balance of utterance amounts between the individual and the group. As an experiment, we develop a system that visualizes the users' degree of talk (how long users speak to whom) in multi-party conversation in real time, and the system scores the talk and gives the users feedback, so that users can control their timing of utterance. In this paper, we introduce the structure and the functions of our system, and then discuss about the comparison results of degree of balancing each user's utterance amount with or without the system.

1. はじめに

「過ぎたるは猶及ばざるが如し」という諺がある。「度が過ぎることは、足りないことと同じくらい良くない」という意味 [1] で、例えば会話においてもこの諺が当てはまる状況は見られる。一般に、複数人の会話では、発話の少ない人と多過ぎる人が存在する。ある人の発話が少ないことによって、その人から得られる情報量が少なくなる。

ある人の発話が多過ぎることによって、そのグループの意見の方向性が、その発話の多い人の意見に偏ってしまう。意思決定の会議や、ブレインストーミングなど、会話参加者が均等に意見を出すことが望ましい場面を想定すると、このような発話の「加減」は会話参加者間で調整されることが必要である。また、外国語のグループレッスンのように、発話をするのがその人の成長に繋がる機会においても、レッスン参加者の発話がほどよい「加減」で行われることが望まれる。著者らはこれまで、複数人の会話において、参加者がいつ他の参加者に話したか、いつ見ていたかを計測し、その累積量と履歴情報を可視化するシステムを開発してきた [2]。本研究では、それに加えて、人の発話加

¹ 立命館大学
1-1-1, Noji-higashi, Kusatsu, Shiga 525-8577, Japan
a) adachi@i.ci.ritsumei.ac.jp
b) seiko@i.ci.ritsumei.ac.jp
c) shimada@ci.ritsumei.ac.jp

減(どのくらいの時間,誰に話しかけているか)を採点し,参加者にフィードバックすることで,各自が次の発話を加減することのできるシステムを開発する.

2. 関連研究

複数人の会話におけるコミュニケーション支援については,これまで様々な方法が研究されている.大島ら [3] は,コミュニケーションの背景にあるダイナミクスを,プロジェクタから投影された光の集合や拡散の表現で可視化するシステムを用い,「会話の場」をデザインする試みを行っている.Terkenら [4] は,小グループのミーティングにおいて,参加者の発話時間や視線の動きを,プロジェクタを用いてビジュアルフィードバックするシステムを提案している.このシステムでは,参加者は視線計測のための反射テープが付いたヘッドバンドと,発話時間計測のためのマイクを身につけなければならない.これらのシステムのように,プロジェクタを用いて情報提示を行う方法は広く用いられている [5], [6], [7].また,フィードバックを行う際に,参加者に直接的な行動の指示はせず,それとなく会話に影響を与えるという特徴を持つシステムもある [8].一方で,Schiavoら [9] は,Kinect センサとタブレット PC から成る,コミュニケーションの流れをサポートするファシリテータのように振る舞い,会話に積極的に関与するようなシステムを提案している.

これらのシステムは,マイクを持ったり身につけたり,プロジェクタが備え付けられた空間が必要であったり,その他特別なものが必要であったりと,会話を始めるまでの準備が大変という問題がある.提案システムでは,タブレット PC で誰が誰に話しているか/誰を見ているかといった計測と可視化の両方を行う.そのため,タブレット PC さえあれば良く,準備が簡単でシステムを使いやすいという利点がある.加えて,可視化表示やメッセージによってユーザにフィードバックするだけでなく,発話の加減に基づく採点結果をフィードバックすることで,ユーザが次の発話を加減することが可能になるという特徴を持つ.

3. システムの概要

本システムは,複数人の会話において,誰が・いつ・どこにいてどちらを向いて発話したかを,タブレット PC を用いて計測・可視化し,自分が話をした時間と相手に話しかけられた時間から会話の得点を算出し,ユーザにフィードバックするものである.

誰が発話したかは,各ユーザが持つタブレット PC の ID から定まる.いつ発話したかは,タブレット PC のマイク入力の音量から判断する.本システムでは音量を計測するのみで,音声認識は行っていない.どこにいてどちらを向いて発話したかは,ユーザの位置と顔の方向から求める.これらの情報から,誰が誰に話しているのか/誰を見てい

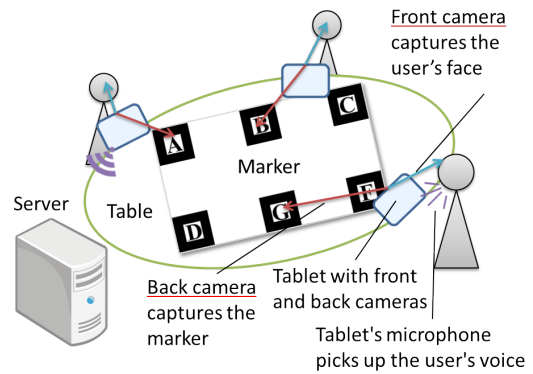


図 1 システムの構成

Fig. 1 System structure.

るのかを推定し可視化する.具体的な計測方法および可視化方法については,4章で詳細を述べる.さらに,複数人の会話では,人により発話の多寡があるため,ユーザが話した時間と他のユーザに話しかけられた時間から,会話の得点を算出し,ユーザにフィードバックすることで,会話にゲーム性を持たせ,会話参加者から均等に意見を引き出すことを狙う.発話加減による得点の採点方法については,5章で詳細を述べる.

図 1 にシステムの構成を示す.本システムでは,各ユーザがタブレット PC を持った状態で,マーカが設置されたテーブルを囲んで会話を行う状況を想定している.ユーザの発話を計測するため,タブレット PC のマイクでユーザの声の音量を計測し,前面カメラで顔検出,背面カメラでマーカ検出を行う.ユーザは前面カメラに自分の顔が映り,背面カメラに複数個あるマーカのいずれかが映るように,タブレット PC を把持しながら会話する.このようにして計測した個々人の発話の情報は,無線 LAN で接続したサーバに送信される.サーバではそれらの情報を統合することで会話全体の特徴量として算出し,各ユーザにフィードバックする.ユーザは会話を行いながら,ときどきタブレット PC の表示を確認することで,発話加減と得点を知ることができ,次の発話を調整することができる.

4. 発話の加減の可視化

4.1 各自の発話の計測

本システムでは,複数人の会話において,どのユーザが,いつどのくらいの時間,どこにいてどちらを向いて話しているのかを計測する.本節ではその計測方法を説明する.

(1) 話しているユーザの特定

ユーザは各自,タブレット PC を把持しており,タブレット PC はサーバに接続され,それぞれに固有の ID がある.現在話しているのがどのユーザかは,この ID によって特定する.

(2) 話している時間の判定

各自のタブレット PC に備え付けのマイクが,ユーザ

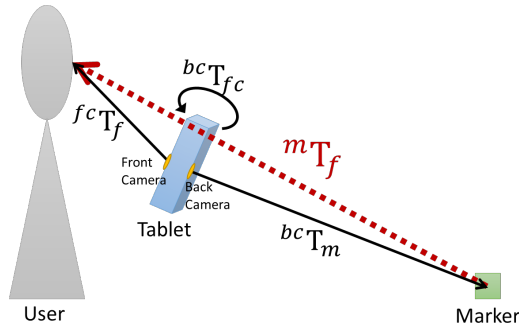


図 2 ユーザ・タブレット PC・マーカ間の位置姿勢関係
Fig. 2 Geometric relation of user-tablet-marker.

の声を拾い、声の音量が閾値を超えると、システムはそのユーザが「発話中である」と判定する。ただし、タブレット PC の前面カメラ映像からの顔検出も併用し、顔が検出されない場合または口の開閉度が低い場合は、声の音量が閾値を超えていても発話中としない。

(3) ユーザの位置と方向の特定

ユーザの位置や顔の方向は、テーブル上のマーカを基準とした世界座標系において、図 2 のように、タブレット PC の前背面カメラからユーザ・タブレット PC・マーカ間の位置姿勢関係を計算することで求めることができる。

上記の (3) について、具体的に説明する。マーカを基準とした世界座標系における顔の座標は、マーカから顔への座標変換行列 $m T_f$ を計算することで求められる。

$$m T_f = (b^c T_m)^{-1} b^c T_{fc} f^c T_f \quad (1)$$

まず、背面カメラからマーカへの座標変換行列 $b^c T_m$ は、背面カメラ画像から検出されたマーカの位置と姿勢を NyARToolkit [10] の機能で取得することで求まる。

また、タブレット PC の前背面カメラはそれぞれ逆向きに備わっており、位置がずれている可能性がある。特定のタブレット PC について、その位置関係は既知であるから、背面カメラから前面カメラへの座標変換行列として、 $b^c T_{fc}$ をあらかじめ定めておくことができる。

次に、前面カメラからユーザの顔への座標変換行列 $f^c T_f$ は、顔の回転成分と並進成分から成る。これらの要素は、前面カメラ画像に対して、オムロンの顔センシング技術 OKAO[®] Vision [11] を用いることによって求めることができる。

このように、タブレット PC の前背面カメラを利用して、映像を補正する研究は他にもあり、例えば Tomioka ら [12] は、前背面カメラの映像を使って、デバイス視点のカメラ画像を利用者視点に補正する研究を行っている。

4.2 会話の特徴量の計測

前節では、マーカを基準として推定されたタブレット PC の ID と位置によりユーザの位置を求め、ユーザの顔の向

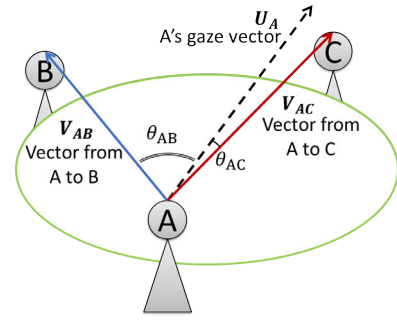


図 3 会話の相手の推定

Fig. 3 Conversational partner estimation.

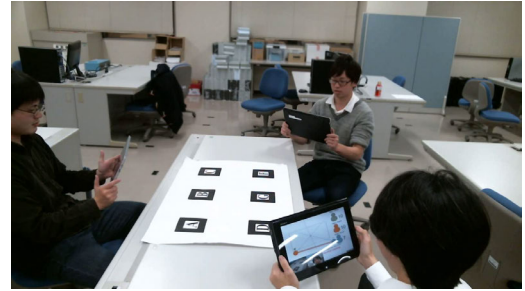


図 4 本システムを用いた複数人の会話状況の例

Fig. 4 Situation example of a multi-party conversation using our system.

きからユーザの方向を特定し、声のセンシングによっていつ・どのくらいの時間、ユーザが話しているかを判定すると説明した。これらの情報を、ユーザが持つ各タブレット PC から収集することで、現在誰が誰に話しかけているのか/誰を見ているのかについての情報、すなわち各ユーザのその時々会話相手をリアルタイムに推定することができる。図 3 に、ユーザの位置と顔の向きを示す。

各ユーザの会話相手は以下の手順で推定する

- (1) ユーザ i の顔の向きとしてベクトル U_i を計算する。
- (2) ユーザ i からユーザ j に向くベクトル V_{ij} を計算する。
- (3) U_i と V_{ij} の類似度 Sim_{ij} は、以下のように計算される。

$$Sim_{ij} = \begin{cases} \frac{U_i \cdot V_{ij}}{\|U_i\| \|V_{ij}\|}, & \text{if } -\frac{4}{\pi} < \theta < \frac{4}{\pi} \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

- (4) Sim_{ij} が最大になるユーザ j をユーザ i の会話相手とする。ただし、 $Sim_{ij} = 0$ のとき、ユーザ i は会話相手を持たない。

このデータを蓄積することで、各ユーザが他のユーザを見ていた時間、または話しかけていた総時間が、会話の特徴量として求められる。

4.3 会話の特徴量の可視化

図 4 に、本システムを用いて複数人で会話をしている様子を示す。図 5 に、タブレット PC の画面上に表示する、会話特徴量の可視化イメージを示す。これは、ユーザ

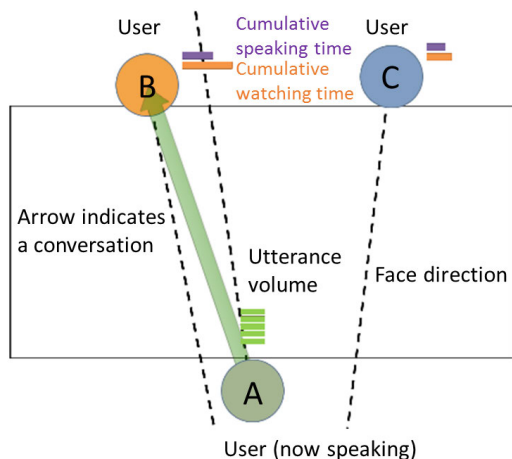


図 5 画面上の会話特徴量の可視化例

Fig. 5 Visualization example of the conversational features on the screen.

A の持つタブレット PC の画面を仮定しており、ユーザ A がユーザ B に話しかけている状況を表している。ユーザは円で表し、顔の方向は点線で表す。ユーザの隣にある横向きの紫色の棒グラフは、ユーザ A からユーザ B への会話量を表し、オレンジ色の棒グラフは、ユーザ A がユーザ B を見ていた総時間を表す。ユーザがこの可視化表示を見ることで、誰が誰にどれだけ話しかけているのかを把握することができる。この機能によって、例えば、それまでに話しかけていない相手に話しかける気持ちを持たせるなどの効果が期待できる。

本研究では、このようにテーブルを中心とした視点を想定しているが、ユーザを中心とした視点での可視化についても検討中である。

5. 発話機会の調整

前節までの手順で計測された発話の加減に基づいて得点を計算し、ユーザにフィードバックすることで、会話にゲーム性を持たせ、会話をコントロールすることを考える。発話の少ないユーザには、より多く発話してもらえよう大きな得点を与え、発話の多いユーザには、他のユーザとのバランスを考慮してもらえよう、話し過ぎると減点されるような採点方式とする。また、会話においては、話すことだけでなく聞くことも重要であるので、相手に話しかけられたときにも得点を得るようにする。本採点方式は、会話参加者全員の発話量を均等にするを想定したものである。それぞれの採点方式の詳細を以下に示す。

自分が話したときの得点

あるユーザ i が話したときの得点 SR_i は式 (3) のようにして算出する。

$$SR_i = \begin{cases} \alpha, & \text{if } t < T \\ \alpha * \left(1 - \frac{N S_i}{\sum_j^N S_j}\right), & \text{otherwise} \end{cases} \quad (3)$$

ここで、 α は話したときの得点比率のパラメータ、 S はユーザの累積発話時間、 t は計測開始時からの経過時間、 T は得点の計算式を切り替える時刻、 N は会話参加者の人数である。つまり、会話の最初の T 時間は発話した分だけ得点になるが、それ以降は、参加者全員の累積発話量の平均に対する自分の累積発話量の割合により得点が決定される。例えば、他の人と比べて発話量が少ない人は多くの得点を得るが、3 人で会話する場合に自分の発話量が全体の $1/3$ を超えている状態で発話すると、その分だけ減点される。これにより、各ユーザが自身の得点を最大化しようと振る舞うことで発話が増減され、参加者全員が均等に発言する方向に向かい、より多くの情報が相互にもたされることが期待できる。

相手に話しかけられたときの得点

相手に話しかけられたときには、一定の得点を得るものとする。すなわち、あるユーザ i が相手に話しかけられたときの得点 LR_i は、得点比率のパラメータを β として、式 (4) のようにして算出する。

$$LR_i = \beta \quad (4)$$

以上から、ある時刻 t におけるユーザ i の会話の得点 $Score_i$ は、これら 2 つの得点の合計を時刻 0 から t まで足し合わせたものとする。

$$Score_i = \sum_{s=0}^t (SR_i^{(s)} + LR_i^{(s)}) \quad (5)$$

この得点をタブレット画面を通じてリアルタイムにユーザにフィードバックし、ユーザはそれを見ながら、各自の発話を調整する。

6. システムの実装

4 章、5 章で述べた原理に基づき、システムを実装した。Windows 8 を搭載したタブレット PC を使用し、アプリケーションは統合型のゲーム開発環境 Unity を用いて開発した。サーバにはサーバサイド JavaScript 環境の Node.js を用い、一般的なデスクトップ PC 上で動作させる。

図 6 に、タブレット PC 上で動作するアプリケーションの実行画面例を示す。画面左上に図 5 で示した会話の特徴量の可視化結果が表示され、画面右上に各ユーザの会話の得点が表示される。画面左下にはタブレットの背面カメラ映像と前面カメラ映像が表示される。背面カメラがマーカを映していれば、検出されたマーカのうち最も信頼度が高いものの ID が表示される。検出に失敗したときには、マーカが見つからない旨のメッセージが表示される。前面カメラの映像についても、顔検出の成否が表示される。

会話の得点は、話したとき / 話しかけられたときに増減するようになっている。得点の表示と一緒に、ユーザ

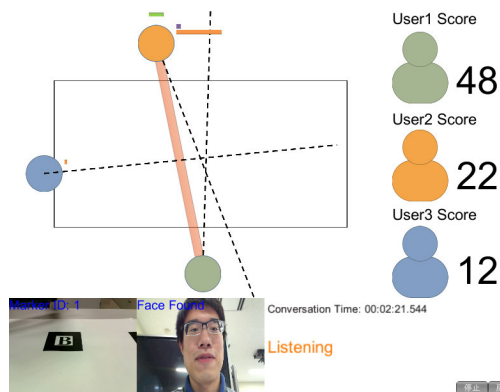


図 6 アプリケーションの実行画面例

Fig. 6 An example of the application screen.

の状態を示すメッセージ (Speaking, Listening, Too much speaking の 3 種類) を画面右下に表示し、ユーザに通知する。また、話しすぎによる減点時と顔見検出時については、画面を見なくてもその状態がわかるように、それぞれ異なる音を鳴らしてユーザに通知し、視覚に依存しないようにした。通知音には、NANO-SOUND [13] で公開されているサウンドデータを使用した。

7. 発話量の均衡化の程度の比較実験

7.1 実験方法

開発したシステムの動作確認として、会話参加者にできるだけ均等に話すよう指示した上で、本システムを用いず普通に会話した場合と、用いた場合とで、発話量がどの程度均衡化されるか、比較実験を行った。被験者 6 名 (著者の所属研究室の学生: 男性 5 名, 女性 1 名, 20-31 歳) を、3 人ずつの 2 グループに分け、実験を行った。実験の様子は録画されており、実験結果の分析に用いた。本実験では、タブレット PC として、Sony VAIO Duo 11 を 2 台と、Microsoft Surface Pro 3 を 1 台使用した。また、式 (3), (4) のパラメータは、 $T = 1$ [min], $\alpha = 0.05$, $\beta = 0.01$ とした。実験の手順を以下に示す。

- (1) 参加者に全員の発話量が均等になるように会話するよう指示する
- (2) 以下の話題についてそれぞれ 5 分間話してもらう
 - 効果的なプログラミングの授業 (システムなし)
 - 作りたいソフトウェア (システムあり)
 グループ 1 は「システムなし」→「システムあり」の順に会話を行い、グループ 2 はその逆順で会話を行う。また、「システムあり」の場合の会話の直前に、システムの表示や会話の得点計測方法について説明する。
- (3) アンケート調査を行う

アンケートでの質問は次の通りである。

Q1. どちらの話題が難しかったか

Q2. システムなしの場合の発言量の調整のしやすさ (-3 ~ +3 の 7 段階評価) とその理由 (自由記述)

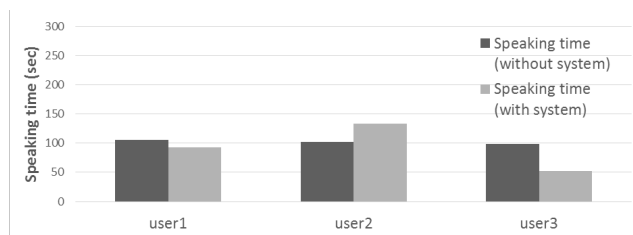


図 7 グループ 1 の各ユーザの発話時間

Fig. 7 Speaking time of each user in group 1.

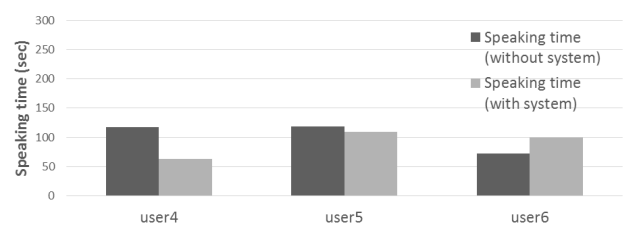


図 8 グループ 2 の各ユーザの発話時間

Fig. 8 Speaking time of each user in group 2.

- Q2. システムありの場合の発言量の調整のしやすさ (-3 ~ +3 の 7 段階評価) とその理由 (自由記述)
- Q3. 感想など (自由記述)

7.2 発話時間の分析結果と考察

グループ 1, 2 の被験者 6 人分の会話について、システムなしの場合とありの場合で、発話時間がどの程度均衡していたか分析した。各ユーザの発話時間は、実験の録画映像と音声を基に著者が計測したものである。

グループ 1 の発話時間を図 7 に示す。システムなしの場合の各ユーザの発話時間はほぼ等しく、発話量の均衡化が自然に行われているようである。一方、システムありの場合については、発話時間のばらつきが見られる。特にユーザ 3 については、システムなしの場合と比べ、発話時間が半減してしまっている。システムなしの場合とありの場合とで話題が異なるため、この話題が当該ユーザが苦手とするものであった可能性はあるが、システムを用いることで必ずしもうまく調整できるとは限らないことがわかった。

グループ 2 の発話時間を図 8 に示す。システムなしの場合は、2 人のユーザが少し話すぎており、残りの 1 人の発話が少ないという結果であった。それに対し、システムありの場合は、先ほど話すぎていたユーザ 4, 5 の発話時間が減少し、ユーザ 6 の発話時間が増加しており、システムが効果を発揮しているように見える。しかし、このユーザ 6 について、システム側で記録された発話量を確認すると、全体の発話量に対しわずか 4.4% しか発話していないことになっている。これは、当該ユーザの顔検出がうまくできておらず、発話が記録されなかったことが原因であり、顔未検出時には音量が閾値を超えても発話として扱わないという設計の影響によるものである。そのため、システムが

提示する会話の得点は正しく算出されておらず、真にシステムの効果で発話量が調整されたとは言えない。

今回の実験では、システムによる発話量の均衡化の効果を明確に確認することはできなかった。ユーザの発話を計測する段階で、顔検出がうまくできないことや、他のユーザの声を当該ユーザの発話として誤判断してしまうなどの問題が影響していたと考える。今後、計測方法を改善した上で、さらに多くの被験者について分析し、システムの効果を確認する必要がある。

7.3 アンケート調査の結果と考察

どちらの話題が難しかったかについては、全ての被験者が「作りたいソフトウェア」の方であると回答した。著者も、この話題の方が多少難しく、システムを適用して発話機会を調整するのに適当であると考え、そのようにした。想定通りの傾向であったが、今回の実験では、システムによる発話機会の調整の効果はあまり確認できなかった。

システムの有無による発言量の調整のしやすさについての回答では、システムなしの場合の平均値が -0.17 ± 1.2 (ややしにくい) であり、システムありの場合の平均値が 0.33 ± 0.9 (ややしやすい) であった。また、調整のしやすさの回答のうち、+1 以上の場合「調整しやすい」、-1 以下の場合「調整しにくい」というように集計すると、システムなしの場合に調整しやすいと感じたユーザは 1 人、調整しにくいと感じたユーザは 2 人であったのに対し、システムありの場合ではそれぞれ 4 人と 2 人であった。このことから、システムがあることで、各自の発話量を調整できたと感じている人の割合の方が高いことがわかった。

感想としては、「点数として自分がどの程度話したかというのが表示されているのは面白かった」「話しすぎの通知を受けることで常に話しすぎと感じたので、発話量は抑えられたと思う」など、システムを用いて発話の加減を採点し、フィードバックすることに好感を持つ意見が得られた。一方で、システムを使用しながら会話することに対し、「タブレットを持ち続けるのはしんどい」「システムを意識することが負担になる」などの意見があった。

アンケートの結果、システムが実際に発話機会を調整できているかどうかに関わらず、発話量の可視化と得点のフィードバックの影響を感じているユーザが多いことがわかった。このようなフィードバックを行うこと自体には肯定的な感触が得られたので、採点方式やそのパラメータを調整することで、より適切なフィードバックが行えるようにしていきたい。また、快適にシステムを使用してもらうために、計測の負担をできるだけ軽減できるよう改良が必要である。

8. おわりに

本稿では、開発した「発話量の可視化と採点に基づき発

話機会を調整するタブレットシステム」について、その構成と機能を紹介し、システムの有無による発話量の均衡化の程度について、比較した結果を述べた。実験の結果、一部ユーザの発話量を調整することができ、アンケートからもシステムの効果を感じているユーザが多いことがわかったが、常にシステムが効果的というわけではなかった。今後は、今回行った比較実験の試行回数を増やし、実験条件を変えながら、より詳しく分析するとともに、会話の量的な変化だけでなく、質的な変化についても調査し、本システムの有効性を確認したい。

謝辞 本研究を遂行するため、オムロン株式会社のご厚意によって OKAO Vision ライブラリを利用させていただいた。ここに、厚くお礼申し上げます。

参考文献

- [1] 故事ことわざ辞典. <http://kotowaza-allguide.com/su/sugitaruwaoyobazaru.html>.
- [2] H. Adachi, S. Myojin, and N. Shimada. Tablet system for sensing and visualizing statistical profiles of multi-party conversation. In *IEEE GCCE 2014*, pp. 407–411. IEEE, 2014.
- [3] 大島直樹, 岡澤航平, 本田裕昭, 岡田美智男. Tabletalkplus: 参加者の共同性や社会的なつながりを引き出すアーティファクトとその効果. ヒューマンインタフェース学会論文誌, Vol. 11, No. 1, pp. 105–114, 2009.
- [4] J. Terken and J. Sturm. Multimodal support for social dynamics in co-located meetings. *Personal and Ubiquitous Computing*, Vol. 14, No. 8, pp. 703–714, 2010.
- [5] T. Bergstrom and K. Karahalios. Conversation clock: Visualizing audio patterns in co-located groups. In *System Sciences (HICSS) 2007*, pp. 78–78. IEEE, 2007.
- [6] Joan M. DiMicco, A. Pandolfo, and W. Bender. Influencing group participation with a shared display. In *Proc. of CSCW 2004*, pp. 614–623. ACM, 2004.
- [7] 藤田和之, 高嶋和毅, 伊藤雄一, 大崎博之, 小野直亮, 香川景一郎, 津川翔, 中島康祐, 林勇介, 岸野文郎. Ambient suite を用いたパーティ場面における部屋型会話支援システムの実装と評価. 電子情報通信学会論文誌 D, Vol. 96, No. 1, pp. 120–132, 2013.
- [8] K. Ogawa, Y. Hori, T. Takeuchi, T. Narumi, T. Tanikawa, and M. Hirose. Table talk enhancer: a tabletop system for enhancing and balancing mealtime conversations using utterance rates. In *Proc. of ACM Multimedia CEA '12*, pp. 25–30. ACM, 2012.
- [9] G. Schiavo, A. Cappelletti, E. Mencarini, O. Stock, and M. Zancanaro. Overt or subtle? supporting group conversations with automatically targeted directives. In *Proc. of IUI '14*, pp. 225–234. ACM, 2014.
- [10] NyARToolkit project — The NyARToolkit project website. <http://nyatla.jp/nyartoolkit/wp/>.
- [11] OMRON Corporation. OKAO Vision | 技術紹介 | オムロン人画像センシングサイト: + SENSING. <http://plus-sensing.omron.co.jp/technology/>.
- [12] M. Tomioka, S. Ikeda, and K. Sato. Approximated user-perspective rendering in tablet-based augmented reality. In *Proc. of IEEE ISMAR 2013*, pp. 21–28. IEEE, 2013.
- [13] III N A N O - S O U N D III. <http://www.nano-graph.com/sound/>.