

2者間におけるコミュニケーションデータを利用した 会話支援システムの試作

隠田亮介^{†1} 真貝晃平^{†1} 多田千里^{†2} 明神学^{†2} 平井佑樹^{†2} 金子敬一^{†2}

仲間とともに問題解決や学習を行う協調学習は、近年の学校教育で行われる学習方法の1つであり、協調学習による豊富な学習効果は教育分野において古くから認識されている。協調学習では、同一の課題について学習者同士が意見交換や合意形成等を繰り返す。それらをスムーズに行うための支援環境が多く開発されている。しかし、協調学習では重要な要素であるはずの学習者同士のやり取り、特に学習者同士の対話状況をトリガとする支援環境は開発されておらず、既存の支援環境では協調学習支援が十分できていない。本研究では、学習者同士の対話状況に注目した協調学習支援システムを開発することを目的とする。本稿では、2者間におけるコミュニケーションデータから会話を支援する一手法を提案し、その手法を実装したシステムについて述べる。

Trial Development of a Conversation Support System using Communication Data between Two Participants

RYOSUKE ONDA^{†1} KOHEI SHINGAI^{†1} CHISATO TADA^{†2}
MANABU MYOJIN^{†2} YUKI HIRAI^{†2} KEIICHI KANEKO^{†2}

Collaborative learning is one of learning methods in modern education and has positive learning effects. A student learns or solves a problem by communicating or negotiating with his/her colleagues in collaborative learning. Numerous collaborative learning support systems have been developed in precedent researches. However, The support systems which have a support function used conversation data among students have not been developed so far. In this study, we aim to develop a conversation support system using communication data among students. We have developed the trial version using communication data between two participants. In this paper, we propose a support method using communication data and the support system.

1. はじめに

協調学習は、同一の課題について学習者が意見交換、競合、交渉、合意形成等を繰り返しグループの合意としての成果を出す。その過程では、個々の意見の調整、修正などが行われる。ここで重要視されるのは、最終的な成果を出すに至るまでのプロセスや、他者との相互作用の質である。最終的に問題が解決されたとしても、そこに至るまでのプロセスによって学習者が獲得する知識やスキルが異なるため、より効果的なプロセスを参加者に提供することを目的として支援環境が設計される[1]。

協調学習を支援する環境は、国内外の研究者によって様々な形で開発されている。近年では、コンピュータに入力したテキスト・チャットを支援に活かすシステム[2]、学習者の視線やジェスチャ等の非言語情報を支援に活かすシステム[3]が開発されている。しかし、学習者同士が意見交換や合意形成等を行うための「対話」の状況を支援に活かすシステムは、筆者の探す限りでは開発されていない。

対話は、協調学習における相互作用の重要なキーとなり、より効果的な学習プロセスを提供するためには、対話を活かした支援も必要である。しかし、この支援を実現するた

めには2つの点で問題である。1点目は音声認識技術に関する問題である。現在の技術では、静かな環境で、想定した声の話者が、想定した範囲の方法で利用する場合には十分な性能を与える[4]ものの、協調学習のような様々な学習者が対話を行う環境では、著しく性能が低下することがある。2点目は開発する支援環境に関する検討を行うためのデータが不足していることである。支援環境を開発する際、どのような支援が有効かということを検討するため、既存研究の結果を利用することはよくあるものの、対話を活かした支援が行われていないため、検討するためのデータが不足している。

本研究では、前述の2つの問題点を考慮し、協調学習において重要な要素である「対話」に注目した協調学習支援システムを開発し、効果的に協調学習を支援することを目的とする。本稿では、2者間におけるコミュニケーションデータから会話を支援する手法を提案し、その手法を実装したシステムについて述べる。

2. 関連研究

平井らは、対面環境における協調学習において、対話が問題解決の成功・失敗にどのような関係があるかを調査した。調査の結果、問題解決に成功した学習者群と問題解決に失敗した学習者群に分けたときに、失敗した学習者群のほうが、1回あたりの発話時間が長く、ある学習者が連続

^{†1} 東京農工大学工学部
Faculty of Engineering, Tokyo University of Agriculture and Technology.

^{†2} 東京農工大学大学院工学府
Graduate School of Engineering, Tokyo University of Agriculture and Technology.

して発話する回数が多いことを示した[5].

この研究成果は、1章で述べた問題点を解決することの根拠となり得る。1章で述べた1点目の問題について、現状では想定した声の話者が、想定した範囲で利用する場合に十分な性能が与えられるが、文献[5]で示した「発話時間の長さ」や「連続して発話する回数が多い」ことは声の話者によらず認識することが可能である。本研究では、このように声の話者によらず認識することが可能な指標を用いて支援することを考える。2点目の問題について、学習者同士のやり取りについて言及した研究[6][7]および学習者同士の対話について言及した研究[5]で示した研究結果は、開発する支援環境に関する検討を行うためのデータとして利用することができる。

3. 会話支援システム

3.1 システムの概要

本システムでは、2者の音声を認識し、音声入力によって得られた内容と、発話全体にかかった時間を取得し発話スピードから、発話者のステータスについて推定を行う。本システムは、大きく「音声入力・音声認識」、「発話情報の閲覧」、「発話ステータスの推定」の3つの機能を持つ。

3.2 音声入力と音声認識

音声入力、音声認識を行うため、高性能音声認識エンジン Julius[8]を用いた。Juliusは京都大学河原研究室をはじめとした研究機関によって製作されているもので、その特徴は主に次の3つである。

- C言語ですべてのソースコードが公開されている[8]
- ディクテーション用のアプリケーション群が公開されている[9]
- モジュールモードに対応しており、他のアプリケーションへの組み込みが容易である

また、Juliusを実際に動作させるためには音声モデルや言語モデルが必要となるが、本研究では配布されているディクテーション用のアプリケーション群にある各モデルを利用することとした。

Juliusには、自身を音声認識サーバとし、TCPを利用して認識結果をクライアントに転送する、モジュールモードという機能がある。このクライアントとして動作するのが jcontrol というプログラムである(図1)。jcontrolはJuliusから送られてくる音声認識結果を表示するだけでなく、コマンド入力によってJuliusの挙動を制御することもできる。なお、Juliusは複数クライアントに対応していない[10].

3.3 発話情報の閲覧

発話情報の閲覧機能では、発話の時刻と発話内容をタイムラインに表示することができる。発話開始時間は現在時刻を取得する time 関数を利用した。また、取得した時間と発話内容の出力をプログラムの適当な箇所に組み込むことで、タイムライン表示を可能とする。

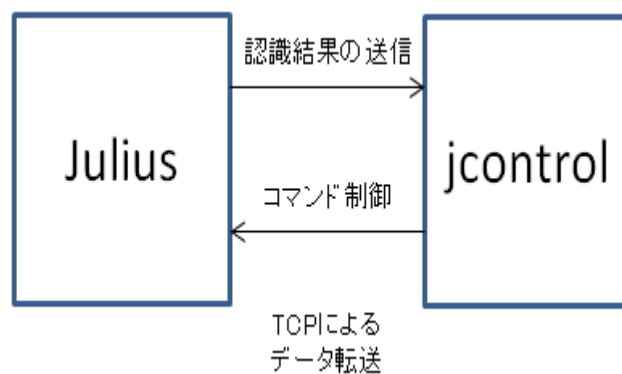


図1 モジュールモードのイメージ図

Figure 1 An Image of Module Mode.

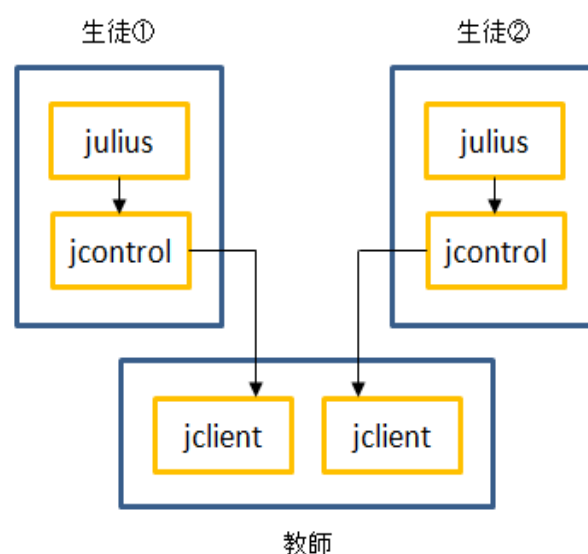


図2 システムのモデル

Figure 2 The System Model.

3.4 発話ステータスの推定

本システムにおける会話支援のための分析では、音声入力によって得られた文字数と、発話全体にかかった時間を利用する。それらによって発話スピードを算出し、それを元に発話者の発話ステータスについて推定を行う。

NHKのアナウンサーの発話スピードが、毎分300~350文字と言われている[11]。一般的な会話はアナウンサーのスピードより少し早いことが想像できる。しかし、このシステムの使用環境では、自分の意見を他者に伝えるという行為が重要であるため、アナウンサーの発話速度を目安に基準を設定することとした。

各閾値の設定については理論的な文献が、筆者が探す限り見当たらず、主観の域を出ないが、アナウンサーの発話速度の前後50文字を基準にした。すなわち、早口であると推定する発話速度を毎分400文字以上、ゆっくりであると推定する発話速度を毎分250文字以下と設定した。

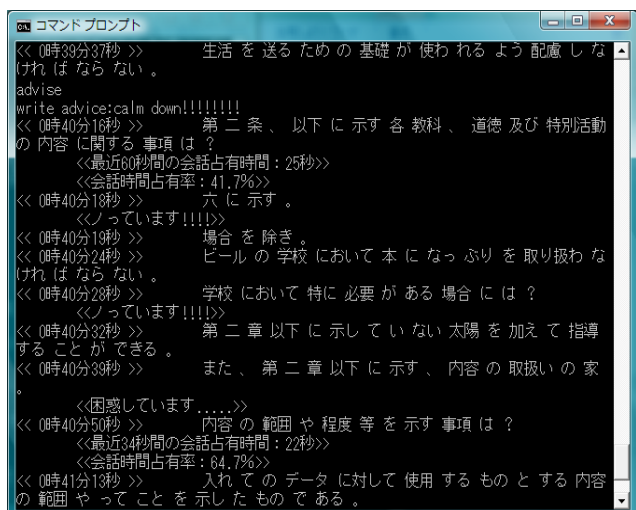


図3 jclientプログラムの実行画面
Figure 3 Execution of a 'jclient' Program.

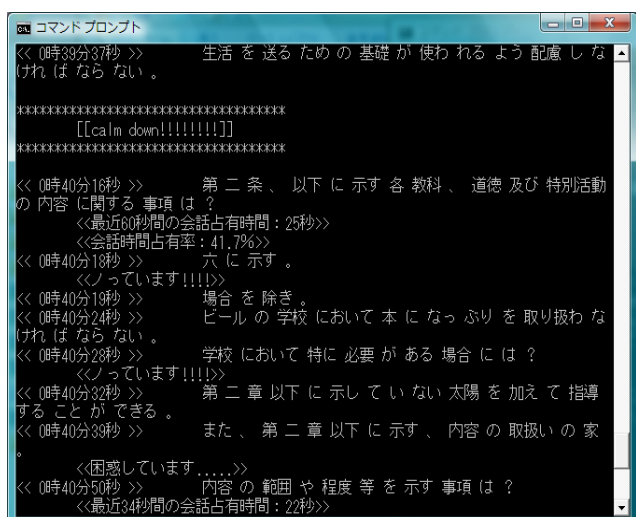


図4 jcontrolプログラムの実行画面
Figure 4 Execution of a 'jcontrol' Program.

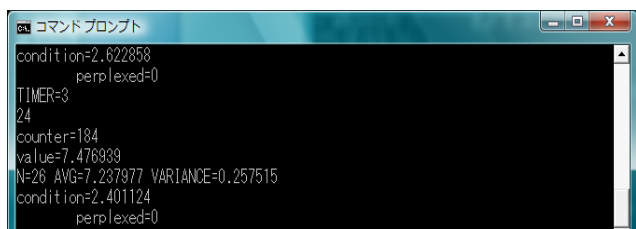


図5 juliusプログラムの実行画面
Figure 5 Execution of an 'julius' Program.

4. システム実装

図2に本研究で実装するシステムのモデルを示す。図2は、システムを教師1名と生徒2名で利用する場合のモデルである。教師と生徒はそれぞれコンピュータを持ってお

り、生徒はそれぞれ音声入力用のマイクを持っている。2名の生徒のやり取りについて、教師がそのコミュニケーションを閲覧でき、教師が会話支援できるようにする。本研究ではJuliusで提供されているプログラムである「julius」、「jcontrol」およびJuliusのプログラム群を参考に作成した「jclient」を用いてシステムを実装した。

4.1 会話スピードの計算

音声認識によって得られた文字列を、ローマ字に変換した文字配列を使いjulius上で計算を行う。文字数は文字配列に含まれる、母音(a, e, i, o, u)、長音(:)、および促音(q)の数によって計上した。これを使い、発話の総時間と文字列によって、発話スピードを求めることができる。3.4節で定めた発話スピードの基準にしたがい、1秒あたり6.7文字(=400(文字/分)÷60(秒/分))を閾値として、それより大きくなる場合は発話スピードが速いと推定するようにした。また、4.2文字(=250÷60)より小さくなる場合は発話スピードが遅いとした。音声入力されたデータから発話スピードを計算し、その結果がクライアントであるjcontrolへ送られる。

4.2 学習機能

前節では、一般的な指標としての閾値を設定し、その値との関係によって発話スピードを推定した。しかし、人の喋るスピードには個性があり、単純に閾値を設定しただけではその個性に対応しきれない。そこで、個々人に固有の閾値を設定するため、統計処理を利用した分析を追加する。

人間の発話スピードには個性があるが、通常時においてはある程度一定であると考えられる。そこで、発話スピードの分布が正規分布であるとの仮定の下、得られた発話スピードが正規分布中のどの位置にあたるかということを利用した検定を行うことにした。このような検定は平均値の検定が適しており、正規分布であると認められる程度の標本が集まった時点で、学習機能が動作するように設定した。この平均値推定によって、個人特有の発話スピードとの比較を行う。有意水準を10%とし、左側検定の対立仮説にステータス「困惑」を、右側検定の対立にステータス「調子が良い」を対応させた。

4.3 詰まりを計測した分析

分析をさらに強固にするために、「詰まりを計測した分析」を実装する。ここでいう「詰まり」とは、発話が短時間で切れてしまったものを指す。話すことに詰まっている時には、一定時間における「詰まり」が多く計測できると考えられることから、比較的短時間での「詰まり」の回数を計測することで、会話に苦しんでいると分析する指標を追加する。具体的には、「詰まり」を検出するために発話の開始を数えるカウンタを利用した。一定時間内にこのカウンタが閾値以上になれば、「詰まり」が多いと判断し、その分析結果を出力するようにした。

4.4 会話時間占有率の計算

本システムでは、会話支援で扱えるパラメータは発話された文字の数とそれにかかった時間のみである。そのため、例えば同じ発話間隔であったときに、言葉に詰まってしまったのか、発話権が移動したのかの具体的な区別がつきにくい。そこで、発話者が主体的に自らの会話を評価できるよう、システム上で会話時間をどれほど占有しているのかを出力できるようにした。

4.5 実行結果

jclient, jcontrol, julius プログラムの実行画面の様子をそれぞれ図 3, 図 4, 図 5 に示す。図 3 のように、行ごとに音声認識した時刻と音声認識の内容が出力される。図 3 の jclient, 図 4 の jcontrol では、同等の時刻と認識結果が得られていることも分かる。一方で jclient では教師によるコメント入力が可能であり、3 行目で先生モード特有のコマンドである“advise”を利用している。ここで入力した文字列が、図 4 の jcontrol 上に整形された状態で出力される。図 5 の julius 実行画面では、counter の値が発話された文字量、それを発話時間で除したものが value として表示されている。これを考慮した結果が、標本数 N, 平均値 AVG, 分散 VARIANCE, 検定式の値 condition と表示される。perplexed の値は 4.3 節で取り上げた「詰まり」の検出のために利用されるものであり、一定時間内に新たな発話が開始された回数を指す。システム実行の結果、ゆっくり発話したときと、速く発話したときに正しく評価されていた。

5. システムの改良に向けて

標本抽出の段階で、発話者がどのような話し方をするかによって、その後の判定が大きく変わってくる。具体的には図 5 では、標本分散が約 0.25 であるため、その後の判定における許容範囲がとても狭くなる。一方、分散が大きくなるよう緩急をつけて発話すれば、許容範囲は広がるが、標本抽出の段階でも制限を設けているため、抽出には限界がある。また、図 5 を見ると、発話した時間は整数時間（1 秒単位）でしか取れていないため、細かい発話速度が得られていない。数秒程度の短い発話においては、同じ時間表示でも発話スピードに幅が生じてしまう。同様に、数秒程度の発話で標本抽出が行われた場合には、一文字あたりの時間の格差が大きく、検定値の幅も大きくなっていくことが分かった。

これらを考慮すると、標本の取り方については、さらに工夫をする必要があると考えられる。これについては、今後検討をしていきたい。

6. おわりに

本研究では、学習者同士の対話状況に注目した協調学習支援システムを開発することを目的とし、2 者間におけるコミュニケーションデータから会話を支援する手法を提案

し、その手法を実装したシステムについて述べた。

現在の音声認識技術では、静かな環境で、想定した声の話者が、想定した範囲で利用する場合には十分な性能を与える。音声認識技術を利用した協調学習支援を実現するためには、この制約を考慮する必要があった。本研究では「発話の時間」や「発話の回数」をトリガとする支援システムの開発を目指しており、これらは想定した声や想定した範囲によらずに認識することが可能である。

今後は、より多くの児童・生徒・学生にシステムを利用してもらい、本システムを改良して、協調学習を本質的に支援する環境の開発を実現したい。

謝辞 本研究の一部は、科学研究費補助金若手研究 (B) 25870207 による。

参考文献

- 1) 稲葉晶子, 豊田順一: CSCL の背景と研究動向, 教育システム情報学会誌, Vol.16, No. 3, pp. 111-120 (1999).
- 2) 林佑樹, 小尻智子, 渡辺豊英: 貢献への気づきを反映した議論支援インタフェース, 情報処理学会論文誌, Vol. 53, No. 4, pp. 1461-1471 (2012).
- 3) Nawahdah, M. and Inoue, T.: Motion adaptive orientation adjustment of a virtual teacher to support physical task learning, Journal of Information Processing, Vol. 20, No. 1, pp. 277-286 (2012).
- 4) 古井貞照: 音声認識技術の実用化への取り組み—音声認識技術の実用化への取り組み, 情報処理学会誌, Vol. 51, No. 11, pp. 1387-1393 (2010).
- 5) 平井佑樹, 井上智雄: ペアプログラミング学習における状態の推定—つまり解決の成功と失敗に見られる会話の違い—, 情報処理学会論文誌, Vol. 53, No. 1, pp. 72-80 (2012).
- 6) Hirai, Y., Inoue, T.: Toward better collaborative problem-solving in programming learning: use of pair programming and its observation, Workshop Proceedings of the 19th International Conference on Computers in Education (ICCE2011), pp. 362-369 (2011).
- 7) 平井佑樹, 井上智雄: ペアプログラミング学習における会話と問題解決の関係, 電子情報通信学会技術報告, Vol. 111, No. 190, pp. 61-66 (2011).
- 8) 大語彙連続音声認識エンジン Julius, <http://julius.sourceforge.jp/>
- 9) 株式会社アドバンスト・メディア「音声認識について」, <http://www.advanced-media.co.jp/introduction/>
- 10) The Julius book, http://julius.sourceforge.jp/juliusbook/ja/desc_module.html
- 11) NHK アナウンサールーム Q&A 「NHK のアナウンサーの話す速度はどれくらいですか」, <http://www9.nhk.or.jp/a-room/qa/>