2006－HI－121 (10)
2006／11／22

テキストチャットを併用する対面口頭議論における チャット発言と口頭発言との時間的関係について

小林 智也
北陸先端科学技術大学院大学
知識科学研究科
t-koba@jaist.ac.jp

西本 一志
北陸先端科学技術大学院大学
知識科学教育センター
knishi@jaist.ac.jp

概要

近年，対面で行われる議論においてチャットを同時併用することが，議論の時間的制約から生じる問題を解決するだけでなく，議論を活発にさせる効果もあるとして注目されている．McCarthy らは，チャットが口頭での対話のバックチャンネルとして機能していると指摘している．この指摘に基づき，筆者らの研究室で実施しているチャット併用のゼミのデータを集め，チャット発言と口頭発言との関係性を調査し，関連あるチャット発言と口頭発言を抽出するアルゴリズムを検討した．その時間的な関係性について調査した．現状では全ての正解集合のうち 20% 程度をカバーする決定木を得ている．

Temporal Relations between Text Messages and Spoken Utterances in a Face-to-Face Communication with a Text Chat

Tomoya Kobayashi
School of Knowledge Science,
Japan Advanced Institute of Science and Technology

Kazushi Nishimoto
Center for Knowledge Science,
Japan Advanced Institute of Science and Technology

Abstract

In recent years, concurrent use of a text-chat system in a meeting has come to gather attention because it can alleviate temporal restrictions of face-to-face communications as well as it can encourage discussions. McCarthy et al. pointed out that there are some temporal or contextual relations between text messages and spoken utterances in a face-to-face communication with using a text chat. Namely, the text chat works as a back-channel of the communication. This paper examines log data of seminars of our laboratory where we use the text-chat system concurrently with oral discussions and discusses an algorithm to extract a spoken utterance related with a text message. Up to now, we have achieved 20% performance of coverage of a correct answer set.

1 背景

近年，対面の口頭議論を活発にするための手法として，口頭議論と平行してテキスト・チャット・システム（以後，チャット）を併用する議論形態が提案されている．この議論形態をケーススタディしたものとしては，研究会 [1]・講義 [2, 3] などがある．

図 1 は，WISS において Comic Chat と呼ばれるチャット類システムを使用した暦本らによる実験 [1] の風景[*1]である．聴衆はネットワーク接続されたノート PC でチャットを行っている．また，右側のスクリーンにはチャット画面が投影されており，発表者はこれをみてチャットの内容を知ることができる．

この報告ではチャットを導入する理由として，議論の時間があらかじめ決められている発表の場で，発表の進行を阻害せずにコミュニケーションをとることのできるチャンネルを提供することにより，発表中でも聴衆同士が議論を行うことを可能にし，発表終了後の質疑応答をより高度な内容のやりとりとすることを，第一に挙げている．前述のケーススタディにおいては，いずれも暦本らと同様の理由によりチャットを導入している．

そのほか，和歌山大学システム工学部デザイン情

*1 出典：http://www.csl.sony.co.jp/person/rekimoto/wiss97/

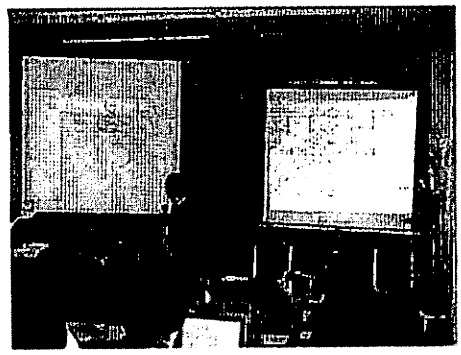

図1　WISS'97 での実施風景

報学科の吉野研究室では，ブレーンストーミングを誘発することを狙い，Remote Wadaman というシステムを使用している．また，創価大学工学部の勅使河原研究室では，協調型テキスト議事録システム [4] を使用し，PowerPoint のプレゼンテーションや音声・映像などの多様なデータを統合したゼミの議事録を分散して作成することによって，会議の記録を個人で作成する場合と比較して負担を軽減することを狙っている．しかしながら，従来，チャット上の発言履歴と口頭での発言を統合し，検索しようとする試みはなされていない．

McCarthy らの報告 [5] ではチャットの利用の形態に関する考察として，チャットはバックチャンネルと同様の性質を持ち，対面でやりとりされている議論に対し，チャット上でやりとりされる発言の内容が時間的・内容的な制約を受けると指摘している．この関係性を活用すれば，チャットから得られるテキスト情報から口頭での発言を検索できるようになると考えられる．そこで本研究では，口頭議論とチャット上での議論の時間的・内容的な関連性について検証し，チャット上の発言を，音声発言の検索用のインデックスとして利用できるかどうかについて検討した．

2　定義

本研究においては，チャットによる会話と，口頭発話による会話が平行して行われる．以下では，チャットでの発言を「チャット発言」，口頭での発言を「口頭発言」と呼ぶ．

チャット発言は，送受信ログ上では特定の時刻に特定の人物によって送信されたメッセージの記録として捉えることができる．しかし，この送受信ログが示すものは送信者がチャット発言を入力し終わった時刻を示すものである．口頭発言との関連性を調査するという観点からは，入力を終えた時刻だけでなく，入力を開始した時刻も必要である．

そこで，データの調査対象としてチャット発言の入力を開始した時間を加え，チャット発言に対して，それを入力するための時間領域として入力開始時刻と入力終了時刻を要素として持つ「入力リージョン」を定義した．口頭発言についても同様に，発話開始時刻と発話終了時刻を要素として持つ時間領域「発話リージョン」を定義した．

2.1　入力リージョン

入力リージョンは，チャット発言を入力するために費やした時間領域を表す．図2に入力リージョンの例を示す．実際の入力リージョンの生成方法としては，チャット・ウィンドウに対するキー入力の履歴から，以下の条件に基づいて作成される．

1. 入力リージョンの単位時間は1秒である．
2. チャット発言の入力欄が空欄の状態から，初めて文字が入力された時刻を入力開始時刻とする．誤入力の訂正も入力時間として含める．
3. チャット発言を送信した時刻を，入力終了時刻とする．
4. キー入力あるいは「削除」機能によって入力欄が空欄になった場合，発言は行われなかったものとして扱い，入力リージョンは生成しない．

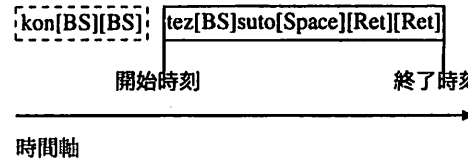

図2　入力リージョンの例

上記の定義でキーの入力ログを解析することによって得られる一連の入力リージョンを，$C(x)(x =$

1,2,3,...) として扱う．また，ある特定の入力リージョン $C(p)$ に対して，入力開始時刻は $C(p)_{start}$，入力終了時刻は $C(p)_{end}$，入力時間は $C(p)_{length}$ として表記する．

2.2 発話リージョン

図3に発話リージョンの例を示す．発話リージョンの分割方法は以下の通りである．

1. 発話リージョンの単位時間は，入力リージョンと同様に1秒とする．
2. 音声が無音でない状態が1秒以上続いたら，無音から有音に変わった最初の時刻を発話リージョンの開始時刻として扱う．
3. 3秒以上無音が続いたら発話リージョンは終了し，無音区間の開始時刻を発話リージョンの終了時刻として扱う．図3においては，1行目一番右のリージョンは，この条件に基づいて作成された．ただし，次に同じ話者が発話する場合もあるので，発話リージョンが終了したからと言って，必ずしも話者交代が発生しているとは限らない．
4. 無音時間が1秒以上3秒以下であっても，別の話者による5秒以上の発話リージョンが同時刻に存在したら，無音になった時刻を発話リージョン終了時刻として扱う．

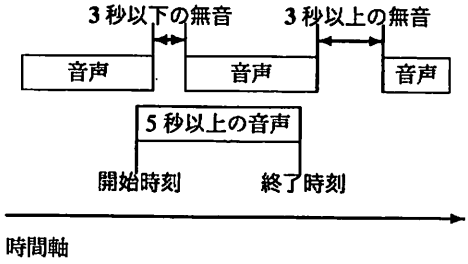

図3 発話リージョンの例

上記の定義で音声データを解析して得られる一連の発話リージョンの配列を，$V(y)(y = 1,2,3,...)$ として扱う．また，ある特定の発話リージョン $V(q)$ に対して，発話開始時刻は $V(q)_{start}$，発話終了時刻は $V(q)_{end}$，発話時間は $V(q)_{length}$ として表記する．

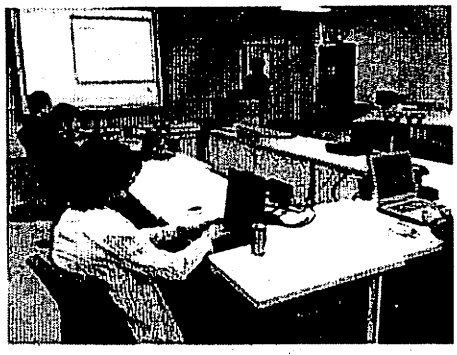

図4 データ収集中の全体風景

3 実験

3.1 データの収集

3.1.1 データ収集の収集元

データの収集は筆者らが所属する研究室のゼミをそのまま記録することによって行った．我々の研究室では，以前から口頭での議論と平行してチャットを併用しており，チャット併用の議論に慣れている．故に，自然な対話データが得られると考えられる．

我々の研究室のゼミは，全員が集合して1名がプレゼンテーションを行う一般的なスタイルで行われている（図4）．参加者は全員ノートパソコンを持ち込み，ノートパソコンはネットワークに接続し，この上でチャットを行う．また，ゼミはペーパーレス化されており，ゼミの発表に使用するプレゼンテーションや，配付資料などがあらかじめサーバにアップロードされている．ゼミの参加者はそれらのファイルにアクセスし，自分のパソコンで閲覧する．これはデータ収集のための特別な環境ではなく，以前から行われてきたゼミの環境をそのまま引き継いでいる．データの収集は，これらの環境に加えデータ収集用にいくつかのソフトウェアおよびハードウェアを追加して行う．

3.1.2 データ収集の方法

データ収集の対象となるのは，以下のデータである．

- 全体で記録するもの

1. 全体の音声録音
2. チャットのログ

- 参加者ごとに記録するもの
  1. キー入力の履歴
  2. 各個人の音声

全体の音声録音は，バラバラに記録される各個人の音声を同期するために必要となる．全体録音は，ゼミの会場にハードディスクレコーダ1台とマイク1本を設置し，ゼミ開始からゼミ終了まで録音することによって記録した．

チャットシステムとしては，マイクロソフト社のMSN Messenger Version 7.5[6] を使用した．チャットのログはMSN Messengerから得られるものをそのまま使用する．あらかじめ参加者用のアカウントと実験者用のアカウントを用意しておき，メンバー登録などを済ませておく．実験開始時には実験者用のアカウントから一斉に参加者をチャットに招待する．実験者は発言しない状態でチャットに参加し続け，受信したメッセージをチャットのログとして記録する．

キー入力の履歴を記録するために「きいろがぁ [7]」を使用した．MSN Messengerのログだけでは入力開始時刻を得ることができないため，キー入力の履歴から入力開始時刻を判定する．ゼミ開始時にきいろがぁを起動し，ゼミ終了時にきいろがぁを終了させ，出力されたログを実験者に送信してもらう．

参加者ごとの音声は，ヘッドセットを配布して録音する．ゼミ中は発表者・聴衆関係なく全員が常時ヘッドセットを着用し録音状態を維持する．ヘッドセットは録音のためだけに使用するため，必ずしも頭につける必要はない．ヘッドセットはノートパソコンに接続し，「超録 [8]」を使用して録音する．ゼミ終了後，録音されたファイルを実験者に送信してもらう．

収集されたデータは，関連付け作業用に整理される．各参加者ごとに記録された音声は，全体録音に沿って同期する．この作業は複数の音声を同時に並べて編集することができる，SonyのVegas[9] という市販アプリケーションを使用して手作業で行った．ゼミでは複数の発表者による複数の内容が取り扱われるため，発表者ごとに位置を区切り，これを実験用の単位とした．全体と同期され，発表者ごとに区切られた音声は，ノイズ除去などの処理を行った後，発話リージョンに変換される．

表1　得られた発表の詳細

ケース	内容	参加者	リージョン 入力	リージョン 発話	時間
ケース1	進捗	9人	72	184	51分
ケース2	進捗	9人	37	45	27分
ケース3	進捗	5人	43	45	23分
ケース4	進捗	5人	54	231	40分
ケース5	進捗	7人	81	65	36分

3.1.3 データの収集結果

これらの作業を行った結果，ゼミ3回分のデータから，発表者5人分の発表を抽出することができた．それぞれのケースの概要は，表1の通りである．

3.2 データ関連付け

3.2.1 データ関連付けの方法

データの収集で得られた5ケース全てに対して入力リージョンと発話リージョンの関連付けを行う．データ関連付けは，各ケースについて最低で3人以上の評価者によって行う．

関連付けは，チャット発言と口頭発言とに関して，意味的・内容的に関係を認められるものをリストアップすることによって行う．関連付け作業のためのツールとして，Voice-Text Association Tool（図5）を作成した．Voice-Text Association Toolは，チャット発言と口頭発言の時間的同期を保った状態で，ゼミの記録を再現する．

①は音声の再生と停止を行うコントロール，②はチャット発言の一覧，③は誰がいつ喋っているかを示すための音声トラックである（図6）．音声トラックは，1行が1人の発話状態を表しており，発話リージョンが存在する領域は黒く，発話リージョンが存在しない領域は白く表示される．音声トラックは30秒分の状態を表示し，中央が現在の再生位置である．④は関連付けボタンであり，選択した関連付けを確定する際に使用する．

Voice-Text Association Toolの関連付け作業は，すべてマウスで行う．主な手順は，以下の通りである．

1. 再生開始ボタン①をクリックし，再生を開始する．
2. 関係を認められるチャット発言と口頭発言を見つけたら，該当するチャット発言②と，該当する発話リージョン③を両方クリックする．
3. 関連付けが認められるチャット発言と口頭発言が選択できたら関連付けボタン④をクリックして，リストに追加する．
4. スピードが速すぎて関連付けができない場合は停止ボタン（■）をクリックすることによって，音声を自由に停止することができる．
5. 音声を停止した後は，音声トラックのスクロール・バーを使って自由な位置から再生を行うことができる．

3.2.2 データ関連付けの結果

以上のデータ関連付け作業をのべ 8 人の評価者に依頼した．評価者はいずれも我々の研究室以外の知識科学研究科の学生であり，関連付け対象のゼミには参加していない．作業時間は，1 時間から 1 時間 30 分で終了した．関連付けの結果を表 2 に示す．表

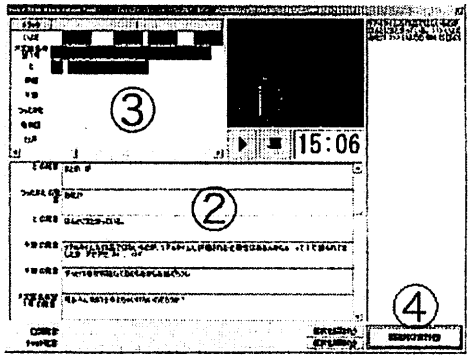

図 5　Voice-Text Association Tool

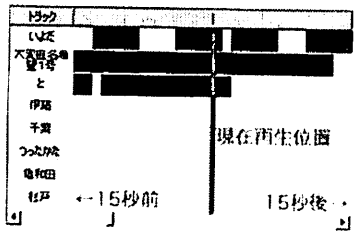

図 6　Voice-Text Association Tool の音声トラック

表 2　データ関連付けの結果

ケース	評価者	組み合わせ総数	関連付け（重複なし）
ケース 1	3 人	13,284	31 関連
ケース 2	3 人	1,665	35 関連
ケース 3	3 人	1,935	48 関連
ケース 3	4 人	12,474	60 関連
ケース 3	3 人	5,265	18 関連

2 における「組み合わせ総数」とは，入力リージョンと発話リージョンの全ての組み合わせパターンの総数である．

3.3 分析

分析では，関連付け作業で得られたケース 1〜ケース 5 までのうち，ケース 1〜ケース 3 までのデータについて，どのような特性があるのかを調べる．ケース 4 とケース 5 は，検索アルゴリズムの検証用のオープンデータとして使用する．今回の実験では，複数の評価者による関連付けの和集合を検索対象とており，3 人の評価者間での一致数による関連付けの切り捨ては行っていない．

入力リージョン $C(p)$ 同士の関係を表す変数，発話リージョン $V(q)$ 同士の関係を表す変数，入力リージョン $C(p)$ と発話リージョン $V(q)$ の間の関係性を表す変数として，以下の値を求めた．また，議論を「プレゼンテーション中」「質疑応答中」の 2 つに分ける状態値と，チャット中で出現回数が 2 回以上の名詞を含むかどうかを示す変数も用意した．

- $C_{\mathrm{length}} = C(p)_{\mathrm{end}} - C(p)_{\mathrm{start}}$
- $V_{\mathrm{length}} = V(q)_{\mathrm{end}} - V(q)_{\mathrm{start}}$
- $CC_{\mathrm{ss}} = C(p)_{\mathrm{start}} - C(p-1)_{\mathrm{start}}$
- $CC_{\mathrm{se}} = C(p)_{\mathrm{start}} - C(p-1)_{\mathrm{end}}$
- $CC_{\mathrm{es}} = C(p)_{\mathrm{end}} - C(p-1)_{\mathrm{start}}$
- $CC_{\mathrm{ee}} = C(p)_{\mathrm{end}} - C(p-1)_{\mathrm{end}}$
- $VV_{\mathrm{ss}} = V(q)_{\mathrm{start}} - V(q-1)_{\mathrm{start}}$
- $VV_{\mathrm{se}} = V(q)_{\mathrm{start}} - V(q-1)_{\mathrm{end}}$
- $VV_{\mathrm{es}} = V(q)_{\mathrm{end}} - V(q-1)_{\mathrm{start}}$
- $VV_{\mathrm{ee}} = V(q)_{\mathrm{end}} - V(q-1)_{\mathrm{end}}$
- $CV_{\mathrm{ss}} = C(p)_{\mathrm{start}} - V(q)_{\mathrm{start}}$

表3　訓練データ（すべてのパラメータを使用）

	T'	F'	合計
T	68	46	114
F	16	16718	16734
合計	84	16764	16848

- $CV_{se} = C(p)_{start} - V(q)_{end}$
- $CV_{es} = C(p)_{end} - V(q)_{start}$
- $CV_{ee} = C(p)_{end} - V(q)_{end}$
- IsInPresentation
  …入力リージョンの入力終了時刻がプレゼンテーション中の入力かどうか（0/1）
- IsIncludeKeyword
  …チャット発言が頻出語を含むかどうか（0/1）

3.3.1　分析方法

上記のデータに対する分析は，Weka[10] を使用して行った．Weka は，ニュージーランドの Waikato 大学が中心になって開発されたデータマイニングアプリケーションである．ケース 1〜3 のデータを用い，J48 クラシファイヤによって決定木を作成した．

3.3.2　分析結果

変数の組み合わせを様々に変更した結果，全ての変数を使用した場合に最もよい決定木を構成できることがわかった．決定木を図 7 に示す．表 3 は，本来の関連（人間が関連付けした組み合わせ）T と決定木により予測された関連 T'，および本来の非関連 F と決定木により予測された非関連 F' を示したものである．訓練データでの評価は再現率 60%，適合率 81% となった．

```
SpeakSpan <= 73
|  CVss <= -68: FALSE
|  CVss > -68
|  |  CVee <= 31
|  |  |  CVes <= 6: FALSE
|  |  |  CVes > 6
|  |  |  |  CVee <= 2
|  |  |  |  |  IsIncludeKeyword = TRUE: TRUE
|  |  |  |  |  IsIncludeKeyword = FALSE
|  |  |  |  |  |  IsInPresentation = TRUE: TRUE
|  |  |  |  |  |  IsInPresentation = FALSE
|  |  |  |  |  |  |  InputSpan <= 5
|  |  |  |  |  |  |  |  InputSpan <= 3
|  |  |  |  |  |  |  |  |  SpeakSpan <= 44
|  |  |  |  |  |  |  |  |  |  CVss <= 10: FALSE
|  |  |  |  |  |  |  |  |  |  CVss > 10: TRUE
|  |  |  |  |  |  |  |  |  SpeakSpan > 44: FALSE
|  |  |  |  |  |  |  |  InputSpan > 3: FALSE
|  |  |  |  |  |  |  InputSpan > 5: TRUE
|  |  |  |  CVee > 2
|  |  |  |  |  IsIncludeKeyword = TRUE
|  |  |  |  |  |  SpeakSpan <= 7: FALSE
|  |  |  |  |  |  SpeakSpan > 7
|  |  |  |  |  |  |  CVee <= 9: TRUE
|  |  |  |  |  |  |  CVee > 9: FALSE
|  |  |  |  |  IsIncludeKeyword = FALSE: FALSE
|  |  CVee > 31: FALSE
SpeakSpan > 73
|  CVee <= -2
|  |  CVes <= -7: FALSE
|  |  CVes > -7
|  |  |  SpeakSpan <= 328
|  |  |  |  CVee <= -280: FALSE
|  |  |  |  CVee > -280: TRUE
|  |  |  SpeakSpan > 328
|  |  |  |  InputSpan <= 16
|  |  |  |  |  SpeakSpan <= 391
|  |  |  |  |  |  InputSpan <= 0: TRUE
|  |  |  |  |  |  InputSpan > 0
|  |  |  |  |  |  |  CVes <= 162: TRUE
|  |  |  |  |  |  |  CVes > 162: FALSE
|  |  |  |  |  SpeakSpan > 391: FALSE
|  |  |  |  InputSpan > 16: TRUE
|  CVee > -2: FALSE
```

図 7　分類学習の出力結果

4　評価

分析結果で得られた決定木をオープンデータに対して適用した．まず，ケース 4 およびケース 5 をあわせたテストデータで評価したところ，再現率は 19%，適合率は 22% となった．

次に，ケース 4 とケース 5 を別々のテストデータとして評価した．ケース 4 のみの結果は，表 5 で，再現率は 16%，適合率は 26%．ケース 5 のみの結果は，表 6 で，再現率は 29%，適合率は 17% となった．

表4　テストデータ（ケース 4 およびケース 5）

	T'	F'	合計
T	14	60	74
F	50	17588	16638
合計	64	17638	17702

表5　テストデータ（ケース 4 のみ）

	T'	F'	合計
T	9	48	57
F	25	12365	12390
合計	34	12390	12424

表6　テストデータ（ケース5のみ）

	T'	F'	合計
T	5	12	17
F	25	5223	5248
合計	30	5248	5278

5　考察

今回のアルゴリズムでは，訓練データに対しては比較的良い性能を達成できているが，オープンデータについてはあまり良い結果を得られなかった．特に，ケース4とケース5では再現率と適合率が逆転したような状態になっている．この原因を調査するために，分析で使用した変数を統合し，位置的特徴に基づいて以下の6つのタイプに分類し，検討を行った．図8〜図13に各タイプを示す．

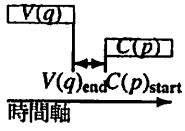

図8　時間的関係性タイプ$T1$

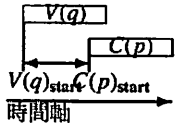

図9　時間的関係性タイプ$T2$

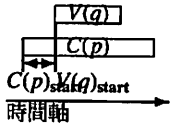

図10　時間的関係性タイプ$T3$

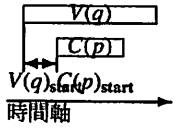

図11　時間的関係性タイプ$T4$

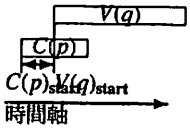

図12　時間的関係性タイプ$T5$

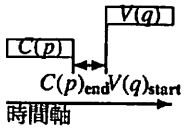

図13　時間的関係性タイプ$T6$

時間的関係性タイプ$T1$は，チャット発言の入力開始よりも前に口頭発言が終了した場合．時間的関係性タイプ$T2$は，チャット発言の入力開始よりも前に口頭発言が開始され，入力中に口頭発言が終了した場合．時間的関係性タイプ$T3$は，チャット発言の入力中に評価対象となる口頭発言が開始され，同じく入力中に口頭発言が終了した場合．時間的関係性タイプ$T4$は，チャット発言の入力開始よりも前に口頭発言が始まり，入力終了後になって口頭発言が終了した場合．時間的関係性タイプ$T5$は，チャット発言の入力中に口頭発言が始まり，入力終了後に口頭発言が終了した場合．時間的関係性タイプ$T6$は，チャット発言が入力終了後に口頭発言が開始された場合をそれぞれ表している．

ケース1〜3において関連付け作業により関連があると判定された組み合わせを時間的関係性タイプに分類し，その分布を表したものが表7である．この分類では，時間的関係性タイプ$T4$に該当する組み合わせが最も多く，次いで時間的関係性タイプ$T1$と時間的関係性タイプ$T6$という順番で分布している．

表7　ケース1〜3の時間的関係性タイプの分布

$T1$	$T2$	$T3$	$T4$	$T5$	$T6$
19.3%	5.3%	7.9%	47.4%	7.0%	13.2%

しかし，ケース4の分布（表8）では，時間的関係性タイプ$T1$が最も多く，次いで時間的関係性タイプ$T6$と時間的関係性タイプ$T4$という順番になっている．訓練データにおける上位3タイプと項目自体は同じであるが，その出現する順番が同一ではない．

表8　ケース4の時間的関係性タイプの分布

$T1$	$T2$	$T3$	$T4$	$T5$	$T6$
47.4%	0.0%	1.8%	12.3%	1.8%	36.8%

また，ケース5の分布（表9）では，時間的関係性タイプ$T4$が最も多く，次いで時間的関係性タイプ$T5$と時間的関係性タイプ$T6$という順番になっている．訓練データにおける上位3タイプに含まれていない時間的関係性タイプ$T5$が含まれている．

表9　ケース5の時間的関係性タイプの分布

$T1$	$T2$	$T3$	$T4$	$T5$	$T6$
5.9%	0.0%	5.9%	58.8%	17.6%	11.8%

決定木に使用されている要素のうち，この時間的関係性タイプに深く関わっているのはCV_{cc}である．

時間的関係性タイプ $T1$〜$T3$ は CV_{ce} が正，時間的関係性タイプ $T4$〜$T6$ は CV_{ce} が負となる．オープンデータでは，時間的関係性タイプ $T1$〜$T3$ と時間的関係性タイプ $T4$〜$T6$ の比は約 1:2 になっているのに対し，ケース 4 は 1:1，ケース 5 では 1:8 と異なっている．この分布の違いの比率が検索結果に影響していると考えられる．

また，時間的関係性タイプ $T1$ および時間的関係性タイプ $T6$ は，入力リージョンと発話リージョンが重ならない場合に該当し，組み合わせとしてはこれらのタイプが最も多くなる．これらのタイプに含まれる関連付けは，ノイズが必然的に多く処理しにくい．そのため，時間的関係性タイプ $T4$ では再現率が低くなっていると考えられる．

6 まとめと今後の展望

本研究では，対面で音声による口頭での会話を行うのと同時並行的に利用されているチャットで意見を交換できる場合に，チャット上で交換されるチャット発言と口頭発言との間にどのような関連性があるかについて検討した．チャット発言と口頭発言を複数の評価者によって関連付けし，発話リージョンと入力リージョンとの時間的関係から，意味的関係を持つ発話リージョンと入力リージョンを抽出するアルゴリズムを構築できるか調査した．

J48 を用いて決定木を作成した結果，訓練データでは再現率 60%，適合率 81% と比較的よい結果を得たが，オープンデータでは再現率は 19%，適合率は 22% とあまり芳しい成果を得られなかった．各ケースごとの時間的関係性タイプの分布の違いに対する処理と，時間的関係性タイプ $T4$・$T6$ におけるノイズの処理に有効な方法を模索し，検索精度を向上させることが今後の課題である．

しかしながら，再現率・適合率という尺度では非常に低い値しか出せてはいないが，1 万通り以上ある口頭発言とチャット発言の組み合わせのなかから，100 個前後しかない正解集合を 20% 弱見つけ出すことができたことを考えると，アプローチとしての正しさを証明していると考える．数学的には，ランダムで 100 個選択した場合，その中に含まれる正解の個数の期待値は 1 個である．アルゴリズムの検証により判明したケース毎の時間的関係性の違いを分析し，その結果を取り入れることで精度を上げることができると考えられる．

参考文献

[1] Jun Rekimoto, Yuji Ayatsuka, Hirotaka Uoi, Toshifumi Arai : Adding Another Communication Channel to Reality: An Experience with a Chat-Augmented Conference, WISS' 98 (1998).

[2] 畠中晃弘，百合山まどか，垂水浩幸，上林彌彦：講義におけるチャットを利用したコミュニケーション促進の実験，情報処理学会研究報告，Vol. 36-11, pp. 61 – 66 (2000).

[3] 百合山まどか，畠中晃弘，垂水浩幸，上林彌彦：チャットを利用した学生間コミュニケーション促進の実験，情報処理学会研究報告，Vol. 37-7, pp. 37 – 42 (2000).

[4] 平島大志郎，田中充，勅使河原可海：協調型テキスト議事録システムの有効性の検討，情報処理学会研究報告，Vol. GN-55, pp. 81 – 86 (2005).

[5] Joseph F. McCarthy, William G. Griswold, danah boyd, Elizabeth Lawley, Elizabeth F. Churchill, Melora Zaner : Digital Backchannels in Shared Physical Spaces: Attention, Intention and Contention, *Proceedings of the 2004 ACM conference on Computer supported cooperative work*, pp. 550 – 553 (2004).

[6] Microsoft : MSN Messenger, `http://messenger.msn.co.jp/`.

[7] sword : きいろがぁ, `http://sword01.web.infoseek.co.jp/key.htm`.

[8] PINO : 超録, `http://pino.to/choroku/`.

[9] Sony Media Software : Vegas / Sound Forge, `http://www.sonymediasoftware.com/products/vegasfamily.asp`.

[10] The University of Waikato : Weka, `http://www.cs.waikato.ac.nz/ml/weka/`.