

Chatplexer：チャットを併用する口頭発表における発表者のための重要発言選択支援の試み

小林 智也^{1,a)} 西本 一志²

受付日 2011年4月18日, 採録日 2011年10月3日

概要：近年、チャットや Twitter などの短いテキストを即時交換できるメディアを対面口頭での発表・質疑と並行させる試みが増えてきている。こうした試みは対面口頭対話のような制限がないチャットというメディアを聴衆に提供することで、より広い視点からの意見をより多く議論に取り込む目的で行われている。しかし、発表者が発表中や質疑応答中にもチャットに注意を払い続けることは困難であり、発表者が重要だと思ふようなチャット発言を議論に取り上げることが難しいという問題があった。本論文では、チャットから対面口頭対話上での話題に対して返信することのできるクロスチャネル返信という概念を提案し、クロスチャネル返信を分析することによって、発表者が重要だと思ふチャット発言を自動的に学習・推定することを試みた。クロスチャネル返信を実装した Chatplexer システムを使用して実験したところ、チャット上の発言の過半数はクロスチャネル返信に対する返信とその子孫ノードであり、発表者が重要だと思ふ発言もそれらのチャット発言であることが多いことが分かった。また、クロスチャネル返信の情報を用いると、発表者が重要だと思ふ発言を J4.8 で学習・推定させた場合に適合率が大きく改善されることが分かった。

キーワード：CMC, チャット, 会議支援, 機械学習

Chatplexer: Supporting Extraction of Important Opinions for a Presenter in an Oral Presentation where a Text-chat is Concurrently Used

TOMOYA KOBAYASHI^{1,a)} KAZUSHI NISHIMOTO²

Received: April 18, 2011, Accepted: October 3, 2011

Abstract: Recently, there have been many attempts that use short-text-exchanging media like chat and twitter concurrently in a face-to-face meeting in order to obtain various opinions from wider viewpoints by more audience. Such media are free from some restricts of the face-to-face meeting, e.g. a rule of turn taking. However, it is actually difficult for a presenter to pay attention on the chat while he/she is presenting. He/she cannot discuss on important chat opinions in his/her presentation. In this paper, we propose a “cross-channel reply” (XCR) that allows the audience to reply the contents of the face-to-face meeting channel from the chat channel. By analyzing the XCRs, we attempt to automatically estimate the important chat opinions using J4.8 decision tree. We implement a novel chat system named “Chatplexer,” which is equipped with XCR function. We conducted user studies using Chatplexer. As a result, we found that more than half of the important chat opinions belong to trees whose root nodes are XCR messages, and that using XCR information the precision ratio is improved.

Keywords: CMC, text chat, meeting support, machine learning

¹ 北陸先端科学技術大学院大学知識科学研究科
Knowledge Science, Japan Advanced Institute of Science and Technology, Nomi, Ishikawa 923-1292, Japan

² 北陸先端科学技術大学院大学ライフスタイルデザイン研究センター

Research Center for Innovative Lifestyle Design, Japan Advanced Institute of Science and Technology, Nomi, Ishikawa 923-1292, Japan

^{a)} t-koba@jaist.ac.jp

1. はじめに

近年、対面口頭での発表や議論と並行して、チャットや Twitter などの短いテキストを即時交換できるメディアを使用する試みが多数行われている（以下では、このような形態の会議を総称して「チャット併用会議」と呼ぶ）。チャット併用会議の最初の例は、暦本らによる WISS '97^{*1}での ComicChat を用いた実験である [1]。以後 WISS では“WISS Challenge” [2] として、対面口頭対話による会議を支援するチャットなどのコミュニケーション・システムを募集し、ワークショップの開催中にこれらのシステムを実際に併用することを継続的に試みている。

このような WISS での取り組みに触発され、通常のチャットを用いたチャット併用会議の実験が国内外で多数実施されてきた [1], [3], [4]。また、授業でチャットを併用する実験も多数行われている [5], [6]。

こうした試みは、より広い視点からの意見をより多く議論に取り込むことを期待して行われている [1]。対面口頭対話では発言権は排他的であり、1人ずつ順番に発言することを強えられる。このため、時間的制約によって質疑や意見を述べる機会はきわめて限られた数の聴衆にしか与えられない。これに対してチャットでは発言権は排他的でなく、複数の参加者が同時に発言することが可能であるため、いつでも自由に発言できる。さらに、チャットは対面口頭対話に対しても独立しているため、口頭で発表中あるいは質疑応答中でも、関係なく発言することができる。したがって、口頭で質問することができない大多数の聴衆にチャットという発言チャネルを与えることによって、発言の数を増加させることが期待できる [1]。実際にいずれの研究においても、チャットは活発に利用され、参加者からも非常に有用であったという意見を得たと報告されている。

一方、学会発表のような発表者と聴衆とが明確に分かれている形式の会議（以下ではこの形式の会議を「プレゼン型会議」と呼ぶ）においては、チャット併用によって聴衆から提出される意見が増える反面、発表者がチャット上での意見や議論を把握することが困難であることが指摘されている [5]。発表者にとって、対面口頭での発表中や質疑応答中にチャットにも同時に注意を払い続けることは、認知負荷が非常に高い。また、チャット上で活発な発言がなされた場合、そのログが長大なものとなるため、短い質疑応答時間中にこれに目を通して重要な発言をピックアップすることは難しい。ゆえに、発表中や質疑応答中に、発表者がチャット上の重要な意見や質問に気づき、それらをその場で取り上げて会議参加者全員で議論することができるようにする支援手段の実現が求められる。

そこで、個々のチャット発言について重要度を付与し、高い重要度を持つ発言のみを選択的に発表者に提供する手段が考えられる。プレゼン型会議には、講義やセミナーのように講師が聴衆に対して知識を教授・伝達することを主目的とするタイプの発表（知識伝達型プレゼン）と、学会発表のように発表者が新規の仮説や知見を発表し、これに対する聴衆からの様々な意見を収集することを主目的とするタイプの発表（意見収集型プレゼン）の2種類が存在する。知識伝達型プレゼンの場合、知識や情報の主たる収集者は聴衆であるから、聴衆の基準によって各チャット発言の重要度を判定すればよい。このためには、これまでに多数試みられてきた、投票などの手段によって聴衆が各チャット発言に重要度を付与する手段が利用できる。一方、意見収集型プレゼンにおいては、聴衆が新たな知識や情報を得られることだけでなく、発表者が「発表者自身にとって重要な意見や情報」を得られることもきわめて重要である。しかしながら、これまで発表者の基準によって各チャット発言の重要度を判定することは試みられていない。

本研究では、発表者がチャット上から発表者自身にとって重要な意見を簡便にピックアップすることを可能とする機能を有する、意見収集型プレゼンでの併用を想定したチャット・メディアである Chatplexer の構築を進めている。これにより、発表者がチャットから取り残されるという問題を解消することを目指している。この実現に向け、本論文では機械学習を用いて、チャットログから発表者にとって重要な発言を、短い質疑応答時間中でも即座に自動抽出する手法を提案し、その性能を検証する。

以下、2章では関連する研究について紹介する。3章では、発表者にとって重要な発言を抽出するための方法を提案する。4章では、提案した手法に基づき実装したシステムの構成について説明する。5章では、実装したシステムを用いた実験について述べ、6章では実験結果を示し、7章では実験結果に基づき提案手法の有効性を議論する。8章はまとめである。

2. 関連研究

従来から、チャット併用会議においてチャット・ログから重要な発言を投票によって抽出する試みはいくつかなされている。backchan.nl [7] や On-Air-Forum [8]、平島らの研究 [9] では、各チャット発言に対して聴衆が投票を行う機能を提供している。backchan.nl では多数の賛同票を集めた記事だけを選ぶことで、発表者が見るべき発言の数を減らして支援している。しかし、これらの試みでは重要性の判断を聴衆が行っているため、必ずしも発表者にとって重要な発言が抽出されるとは限らない。たとえば Backchan.nl では、「お腹がすいた」などのそのときどきの聴衆の感情を表す発言がランキングの上位になることがあったと報告されている。

^{*1} 日本ソフトウェア科学会主催 Workshop on Interactive Systems and Software 1997.

チャット併用会議での使用を意図したものではないが、由井蘭ら [10] は、チャットの各発言に対して「質問」「回答」などの意味タグを付加するセマンティックチャットを提案し、これによって各発言の意図を明確にする試みを行っている。しかし、この試みにおいても発言への意味タグを付与するのは各発言の発言者であるため、これをプレゼン型会議に併用した場合、やはり重要性などの判断は聴衆が行うことになる。

On-Air-Forum [8] においては、スライドのページと座標を指定してコメントを送信することができる。つまり、各コメントはスライド上に記述されているコンテンツと密接にリンクするものとなる。しかし、スライド上にすべての発表内容が記載されているわけではないため、口頭による発言内容に対してもコメントできることが必要である。

以上のように、従来のチャット併用会議におけるチャット上の発言の重要性は聴衆が判断するものばかりであった。また、発表者が口頭で話した内容に対してコメントを明示的に付与する手段を提供する試みは、我々の知る限り存在しない。本論文では、次章で提案する手法により、スライドだけでなく発表者が口頭で話した内容に対してもコメントを明示的に付与することを可能とし、さらにこの返信関係情報をもとに、チャット上の発言の重要性を発表者の基準で推定することを目指す。

3. 提案手法

McCarthy らは、チャット併用会議を観察・分析した結果、対面口頭対話と並行して行われるチャットは、バックチャンネルに類似しており、対面口頭での議論から時間的・意味的制約を受けると考察している [11]。この知見に基づき、筆者らは、チャット併用会議においては対面口頭での対話内容とチャット発言との間の対応関係を明確化し、これを記録することが重要であると考えた。さらに発表者の口頭での発言に対する返信であれば、それは発表者にとっても重要である可能性が高いと考えられることから、この対応関係を記録し分析を行うことによって、発表者にとって重要なチャット発言を推定できると考えた。この分析によって、発表者にとって重要な発言を抽出するアルゴリズムを実現できれば、時間的制約がある質疑時間においても、チャット上の有益な意見を取り入れて議論を進めることが容易になると期待される。そこで本論文では、対面口頭での対話内容とチャット発言との間の対応関係を明確化し、記録するための手段として、「クロスチャンネル返信」という概念を提案する。

3.1 クロスチャンネル返信

チャット併用会議において用いられるメディアは、音声(口頭)、スライド、チャットの3つである。音声とスライドは対面口頭対話として1つのチャンネルを形成し、会議の

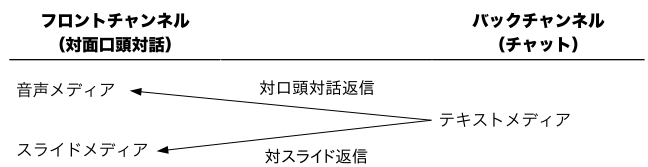


図 1 メディアおよびチャンネルと返信の関係

Fig. 1 Relation between media (channel) and response.

主となる話題が扱われるフロントチャンネルになる。これに対してチャットは、単体で1つのチャンネルを形成し、バックチャンネルとして機能する。クロスチャンネル返信は、フロントチャンネルの対面口頭対話の内容に関連づけて、バックチャンネルのチャットから返信する機能である。具体的には「口頭で発言中の内容に対する返信」(以後、対口頭対話返信と呼ぶ)と、「スライドに対する返信」(以後、対スライド返信と呼ぶ)の2種類の返信機能を提供する。

たとえば、発表者が現在口頭で説明中の内容に対して何らかの疑問や意見を持った際には、口頭で発言中の内容に対する返信としてチャット発言を送信する(対口頭対話返信)。また、発表者がこれまでに表示したプレゼンテーションのスライドに対して何らかの疑問や意見を持った際には、スライドの内容に対する返信としてチャット発言を送信する(対スライド返信)。この2種類のクロスチャンネル返信によって、バックチャンネルからフロントチャンネルへ、チャンネルを越えた明示的な返信が可能になる。この関係を以下の図 1 に示す。

なお、クロスチャンネル返信としては、対面口頭対話(フロントチャンネル)からチャット(バックチャンネル)への返信も考えることができる。これについては、発表者が発表者自身の観点によってチャット上から重要な発言を取り上げることができないという、本論文で取り扱う問題が解決したあとに検討する予定としているため、本論文では対象としない。

4. システムの構成

クロスチャンネル返信機能を実装したチャット・システム「Chatplexer」を開発した。Chatplexerは、新規発言と通常の返信に加え、クロスチャンネル返信を入力・保存および表示することができる。システムの全体像を図 2 に示す。図中の「スライド発言」とは、発表者のプレゼンに用いているスライドを画像としてチャット(聴衆)に配信したものである。スライド発言の詳細については「発表者用クライアント」の項で説明する。

4.1 サーバ

サーバは PHP 5 モジュールを組み込んだ Apache HTTP サーバ 2.2, MySQL 5.1 データベースサーバを用いた。後述する聴衆用システムで低遅延でのメッセージ配信を実現

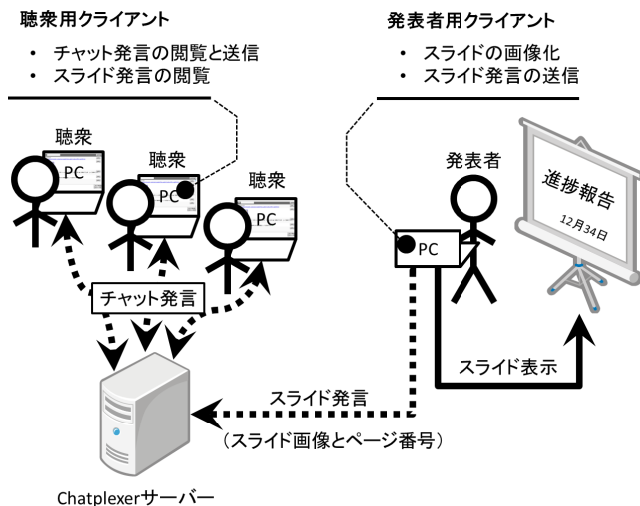


図 2 システムの全体像

Fig. 2 Systems overview.

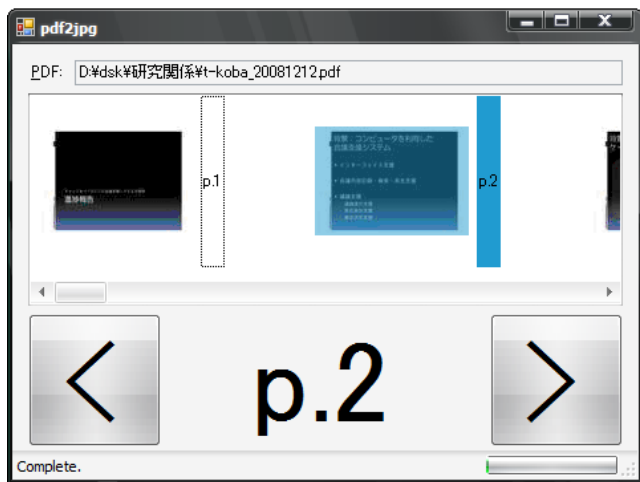


図 3 発表者用クライアントの外観

Fig. 3 Image of client for presenter.

するため Long-Polling 技術を使用するので、Apache および MySQL は多数の接続を高速に受け付けるよう設定を調整した。

Chatplexer では、聴衆が投稿する通常の文字列での発言 (チャット発言) のほかに、後述するように発表者が現在表示しているスライドを示すスライド発言も扱う。サーバは、これらの発言に対して正の整数による一意のシーケンス番号を割り当ててデータベースに記録する。シーケンス番号は、最初の発言が 1 であり、発言が投稿されるごとに 2, 3 とカウントアップする。

4.2 発表者用クライアント

発表者用クライアント (図 3) は、発表者がスライドを表示して発表を行う際に用いるアプリケーションである。発表者用クライアントは、Windows フォームアプリケーションとして C# で実装した。スライドは Portable Document Format 形式 (PDF 形式) で用意するものとした。発表者



図 4 聴衆クライアントの外観

Fig. 4 Image of client for audience.

用クライアントが利用者 (発表者) に対して提供する機能は、スライド用 PDF を取り込んで画像に変換する機能、スライド (ページ) を全画面表示する機能、スライド (ページ) を切り替える機能、の 3 つである。

発表者用クライアントは、発表を開始するとスライドを画像に変換して全画面表示する。スライドの切替え時には、これから表示しようとするスライドの画像がすでにサーバにアップロード済みであるか (すなわち、チャット・ログ上に 1 度でも表示されているか) どうかを確認する。アップロードされていない場合は、オープンソースの PDF 描画プログラム mupdf によりスライドの該当ページを JPEG 形式の画像に変換し、JPEG ファイルをサーバにアップロードした後、ページの切替えをサーバに通知する。アップロード済みである場合は、スライドの ID のみをサーバに送信し、ページが切り替わったことだけを通知する。サーバは、発表者用クライアントからスライド切替え通知を受け取ると、対応するスライド画像をスライド発言として聴衆用クライアントに配信する。すなわちスライド発言とは、スライドの切替えをトリガとして発言される、そのとき表示されているスライドの画像を、チャット発言として表示したものである (実際の表示例は聴衆用クライアントの項で示す)。

4.3 聴衆用クライアント

聴衆用クライアント (図 4) は、聴衆がチャットの表示と投稿に使用するアプリケーションである。聴衆用クライアントはウェブアプリケーションとして PHP と JavaScript で実装した。Long-Polling 技術と Ajax 技術を使用しており、発言の配信遅延は 1 秒以内である。聴衆はブラウザに聴衆用クライアントの URL を入力することによって利用する。

チャット・ログには、聴衆が送信する発言と、発表者用クライアントから自動投稿される、現在発表中のスライドのサムネイル画像を発言本文とするスライド発言が表示される。いずれの発言についても、シーケンス番号と送信者名があわせて表示される。ただしスライド発言の送信者名は「発表者」となる。新しい発言が追加されるとチャット

は自動的に新しい発言までスクロールする。ただし発言入力欄に文字列を入力中である場合は、チャット・ログの表示範囲が変更されることを避けるため、入力中の聴衆用クライアント上では自動スクロールしない。

聴衆用クライアントからの発言送信方法としては、新規発言、通常返信、対スライド返信、対口頭対話返信の4通りがある。いずれの方法についても、本文や送信者名が空欄であると警告が表示され、チャット発言を投稿することはできない。また、間違えて送信してしまうことを避けるために、キーボード操作（主に Enter キー）による投稿はできないようになっている。

また、進捗報告発表は前の発表者の質疑応答が終了した後、即座に次の発表者が発表を開始する場合がある。この場合、チャットの発言を入力し終わる前に次の発表者が発表を始めてしまうと、単純なシーケンス番号や入力終了時刻（送信時刻）だけではチャット発言を発表者ごとに正確に分けることができなくなってしまう。それを考慮し、すべてのチャット発言について、発言本文を入力し始めてから送信されるまでに経過した秒数を取得し、発言送信時刻とあわせてサーバに蓄積している。ただしこれらの送信時刻や入力にかかった時間などの時刻に関する情報は聴衆用クライアントの画面上には表示せず、返信関係構造の解析と発表者ごとのチャット・ログ分割など、分析にのみ使用する。また、通常返信、対スライド返信、対口頭対話返信のいずれについても、1つの返信発言中に含むことのできる返信タグ（後述）は1種類・1つのみに制限した。

以下、各発言の送信方法について詳述する。

4.3.1 新規発言

新規発言は、先行するいずれの発言も参照しない、新たな話題の最初の発言を送信する際に使用する。一般的なチャットであればこのような発言も多数なされうるが、本論文が対象としているプレゼン型会議で並行して行われるチャットでは、このような発言が生じる頻度は少ないと予想される。このため、図4に示すように、インタフェースの設計においては新規発言ボタンを発言入力欄から遠いところに配置した。

なお、後述する通常返信やスライド返信で使用する返信タグ「>> n」や、対口頭対話返信で使用する返信タグ「>> *」を、新規発言に手動で入力して返信発言としてしまうことを防止するために、これらのタグが含まれていると警告を表示し、投稿できない仕様とした。

4.3.2 通常返信

図4に示すように、チャット・ログ中の各発言の表示部の右端に「返信ボタン」が表示されている。入力欄に発言内容を入力した後、返信したい発言に付与されている返信ボタンを押すことで、その発言の返信として発言が送信される。送信された通常返信発言の末尾には、返信タグ「>> n」（nは返信対象発言のシーケンス番号）が自動的に

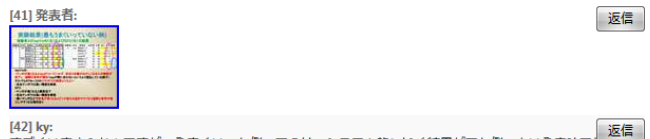


図5 スライド発言の例
Fig. 5 Example of slide-message.

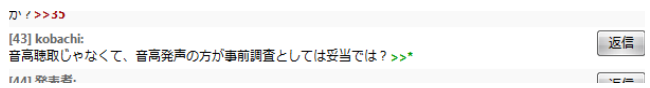


図6 口頭発言への返信の例
Fig. 6 Example of response to current speaking.

付加される。この返信タグには、返信対象発言へのリンクが自動設定されているので、このタグ上にマウスカーソルを重ねることで、返信対象発言を遡ってトレースしていくことができる。

4.3.3 対スライド返信

発表者用クライアントから自動投稿される、スライドのサムネイル画像を発言内容とするスライド発言にも、聴衆からの発言と同様にシーケンス番号が付与され、さらに「返信ボタン」も付与されている。図5に、スライド発言の例を示す。

チャット・ログ上にはすべてのスライド発言が残っているので、過去に遡って任意のスライドに返信することも可能である。入力欄に発言内容を入力した後、返信したいスライド発言に付与されている返信ボタンを押すことで、そのスライド発言の返信として発言が送信される。送信された対スライド返信発言の末尾には、返信タグ「>> n」（nは返信対象であるスライド発言のシーケンス番号）が自動的に付加される。この返信タグには、返信対象発言へのリンクが自動設定されているので、このタグ上にマウスカーソルを重ねることで、返信対象であるスライド発言を遡ってトレースすることができる。

4.3.4 対口頭対話返信

対口頭対話返信は、チャット上には存在しない、口頭での対話内容に対する返信である。口頭対話は音声データであるため、過去の発言内容について厳密に返信先を指定させることは難しい。そこで Chatplexer では「今喋っている内容」だけに限定して対口頭対話返信できるものとして、対口頭対話返信の入力が負担にならないようにした。対口頭対話返信の発言方法は、発言入力欄に返信内容を入力した後、発言入力欄のすぐ右にある「今の口頭会話にツッコミ」ボタンを押すだけである。対口頭対話返信を表すタグとしては、通常返信タグを変形した「>> *」という表記を用いた（図6）。この「>> *」を対口頭対話返信タグと呼ぶことにする。対口頭対話返信では、入力が終了して送信した時刻ではなく、チャット入力欄に入力を始めた時刻に対口頭対話返信を行ったと見なす。

5. 実験

実験ではまず、Chatplexer を使用して意見収集型プレゼンを実施し、クロスチャネル返信のデータを収集する。発表終了後、発表者に対して Chatplexer 上で行われたチャットのログを提示し、自分にとって重要と判断される発言をピックアップしてもらう。こうして得られたクロスチャネル返信データと重要発言のデータを用いて、機械学習によって重要発言の推定を試み、その性能を検証する。

5.1 実験の手順

Chatplexer を 2010 年の 11 月～12 月の 2 カ月間にわたって、筆者らが所属する研究室のゼミ 4 回で使用した。メンバは 1 名の教員（本論文第 2 著者）、5 名の博士課程学生（うち 1 名は本論文第 1 著者）、修士課程学生 11 名で行われた。修士課程学生のうち 3 名が女性（うち 2 名は中国語が母語）で、それ以外は全員男性（うち 1 名は中国語が母語）である。実験期間中に欠席や見学者によって若干の参加者変動がある。

進捗報告ゼミでは、1 回のゼミにつき研究室のメンバ 4～6 名が進捗報告を行う。進捗報告は意見収集型プレゼン形式で実施される。発表予定者は 1 人ずつ順に登壇し、まず現在取り組んでいる作業の内容と進捗状況について、前回の進捗報告からの差分だけではなく、外部発表と同様に背景、目的などを含み研究全体が把握できるように発表する（図 7）。その後、聴衆を交えて口頭での質疑応答と意見交換を行う。Chatplexer は、この発表と質疑応答の間を通じて使用される。

発表者の観点に基づく重要な発言を得るために、ゼミ終了後に発表を行った発表者にチャット・ログを印刷したものを渡し、意見や質問として自分自身の研究にとって重要であると思われるチャット発言をピックアップしてもらった。ただし、重要さを段階的に重み付けすることは求めず、重要か否かだけの判断を求めた。このピックアップ作業は十分に時間的余裕を与えて行った。

Chatplexer のログは、人間が投稿したチャット発言と、発表者用アプリケーションが自動的に投稿したスライド発

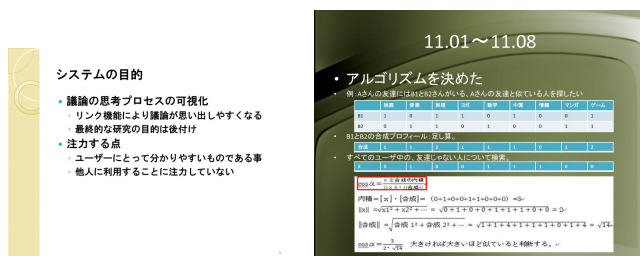


図 7 進捗報告スライドの例

Fig. 7 Example of a slide page from progress report presentation.

言が混在した状態のものとなるが、以降の分析はすべて、人間が投稿したチャット発言のみを対象としたものである。

5.2 収集したデータの概要

4 回の進捗ゼミで、15 名の発表者によって計 20 回の発表が行われた。全部で 2,496 発言のチャット・ログと、248 発言の重要発言が得られた。得られた重要発言から、自動車内の会話を研究している被験者が選んだ重要発言の例を表 1 に示す。発言種別ごとの内訳はクロスチャネル返信が 510 発言（20.4%）、新規発言が 312 発言（12.5%）、通常返信が 1,674 発言（67.1%）であった。括弧内は内訳である。

得られたチャット・ログは、入力開始時刻を用いて発表者ごとに分割した。平均すると、発表 1 件あたり 10 分 17 秒の発表を行いその後 27 分 1 秒の質疑応答をしており、全体で 37 分 18 秒であった。その約 37 分間に平均 124.8 発言のチャット発言が送信された。1 分あたりの平均発言数は、発表中は 3.4 発言/分、質疑中は 3.2 発言/分、全体では 3.3 発言/分であった。チャット発言全体のうち重要発言が占める割合（重要発言占有率）は、発表中が 9.3%、質疑中が 10.6%、全体では 9.9%であった。

実験にあたって質疑応答に使用するチャネルは制限していなかったが、発表者は口頭およびチャットの両方からの質問に対してほとんど口頭で回答し、チャットを見て質問を取り上げることはあっても、質問への回答をチャットで行うことはほとんどなかった。質疑応答中に発表者が発言をチャットに投稿したのは 1 発言のみであった。

チャット発言の返信関係は、メールのスレッドと同様に木構造をなしている（図 8）。クロスチャネル返信発言をルートとし、これに対する通常返信で構成される構造木（クロスチャネル返信木）と、新規発言をルートとして、これに対する通常返信で構成される構造木（新規木）を考える。意味的に見れば、クロスチャネル返信木はフロントチャネルである口頭対話あるいはスライドの内容に対する返信お

表 1 重要発言の例（いずれも同一発表者）

Table 1 Example of chat messages marked as important for presenter by own self.

発言種別	内容
新規発言	聖地巡礼ツアー
対スライド返信	あ、ほんとにごっこ遊びにするのか >> 51
対口頭対話返信	目的が観光なのかグルメなのか >> *

- [542] zawareo: 重要発言数ってメモボチられた発言? >>*
- [544] 田中: これ言っているのが解らないですけど この前のアンケートで >> 542
- [546] zawareo: なるほろ >> 544
- [545] ふじた: 発表者による評価ではないでしょうか 終わった後のアンケートの >> 542
- [548] ky: 重要発言は本当に重要なのか >>*
- [553] ky: まあ、発表者から見た場合だから正しいのか >> 548
- [554] Kazz: だど >> 553
- [557] 田中: 発表者が質疑応答時に取り上げたい発言=重要発言だったかと >> 553
- [558] ちは: まったくなかったら悲しいな...どきどき >> 557

図 8 返信による木構造の例

Fig. 8 Example of response-tree.

よび意味的に連鎖する発言によって構成され、新規木はフロントチャンネルとは無関係な内容の発言および意味的に連鎖する発言によって構成されている。

クロスチャンネル返信と新規発言がそれぞれの構造木のルート・ノードとなるので、クロスチャンネル返信木は510本(62.0%)、新規木は312本(38.0%)である。発言単位では、対口頭対話返信をルート・ノードとするクロスチャンネル返信木に1,336発言、対スライド返信をルート・ノードとするクロスチャンネル返信木に167発言、合計1,503発言(60.2%)がクロスチャンネル返信木に属している。新規木に属するのは993発言(39.8%)であった。

発表1件あたりでは平均で54.8本の返信木があった。クロスチャンネル返信木の数、ならびにクロスチャンネル返信木に属する発言の数は、いずれも過半数を占めており、チャット上ではフロントチャンネルである口頭対話の内容に関連する発言が多くやりとりされていることが分かる。

クロスチャンネル返信木に含まれる重要発言と新規木に含まれる重要発言を比較すると、211発言(85.1%)の重要発言がクロスチャンネル返信木に属しており、うち対口頭対話返信をルート・ノードとするクロスチャンネル返信木は184発言、対スライド返信をルート・ノードとするクロスチャンネル返信木に27発言が属している。新規木に属する重要発言は37発言(14.9%)であった。重要発言占有率で見ると、クロスチャンネル返信木の重要発言占有率が平均14.0%であるのに対し、新規木の重要発言占有率は3.7%であった。このように、重要な発言は新規木にはあまり存在せずクロスチャンネル返信木に多く存在しているが、クロスチャンネル返信木の数自体が多いため、クロスチャンネル返信木における重要発言占有率が大きく高いというわけではない。

6. 重要発言推定実験

実験データに含まれる返信木の情報をもとに、重要な発言を推定できるかどうかを検証した。推論アルゴリズムとしては、決定木を用いる。モデルの作成と検定にはWeka 3.6.4を使用した。Wekaには、J4.8(C4.5の改良版)やSupport Vector Machine(SVM)、単純ベイズ分類器など、機械学習で用いられる一般的なアルゴリズムがあらかじめ搭載されており、データを用意するだけで機械学習による分析を始めることができる。データはAttribute-Relation File Format(ARFF)という属性値情報を含んだWeka専用形式に変換してWekaに読み込ませた。決定木はJ4.8アルゴリズムを使用して作成する。

本論文では、文法構造や言葉の意味には立ち入らず、主として返信の形態に基づいた重要発言の推定を試みる。クロスチャンネル返信を含め、返信の形態が言語によって異なるとは考えにくい。ゆえに本手法によって、言語種別に依存しない重要発言推定が可能となると考えられる。そこで、返信形態以外のパラメータとしては、やはり言語種別

表 2 3つのパラメータの組合せによる推定精度の比較

Table 2 Comparison of accuracy of assumption among three combinations of the parameters.

組合せ条件	適合率	再現率	F 値
Full-XC 条件	61.5%	12.9%	0.213
Slide-XC 条件	28.6%	1.6%	0.031
No-XC 条件	28.6%	1.6%	0.031

に依存しない、発言の文字数とURLを含むかどうかというパラメータのみを採用した。URLを含むかどうかは本文中に「http://」か「https://」が含まれるかどうかを用いて判定する。URLは外部資料への参照であり、発言の重要度に大きく関わると考えられる。

学習に用いたすべてのパラメータを以下に示す。

- 言語に依存しない基本のパラメータ
 - 発言の文字数
 - 発言がURLを含むかどうか
- 返信関係の構造木に関するパラメータ
 - 発言の構造木における階層深さ
 - 発言に対する返信(子ノード)の数
 - 発言が属する構造木全体のノード数
- クロスチャンネル返信に関するパラメータ
 - クロスチャンネル返信ではない
 - 対口頭対話返信
 - 対スライド返信
- その他のパラメータ
 - 発言の入力にかかった時間

これらのパラメータを、次に示す3通りの組合せ条件に分けて学習を行う。

Full-XC 条件 クロスチャンネル返信に関するパラメータを含む全パラメータを使用して学習を行う。

Slide-XC 条件 On-Air Forum[8]などでは、スライド上に発言を書き込む機能を有している点で、対スライド返信機能が実現されていると見ることができる。そこで、対スライド返信のみをクロスチャンネル返信と見なし、対口頭対話返信はクロスチャンネル返信ではないと見なして学習を行う。

No-XC 条件 クロスチャンネル返信に関するパラメータをいっさい使用しないで学習を行う。

最終的に発表者がピックアップした重要発言と、学習に基づく推定結果を比較し、これら3つのパラメータの組合せの性能を比較する。なお検定は、取得したデータを10分割してそれぞれを1回ずつテストデータとする10-fold交差検定により行った。結果を表2に示す。

表2から分かるように、Full-XC条件で他の2つの組合せよりも高い精度を達成し、Slide-XC条件ではNo-XC条件と同一の精度である。Slide-XC条件とNo-XC条件が同じ精度となったのは、学習の結果出力された決定木が同一

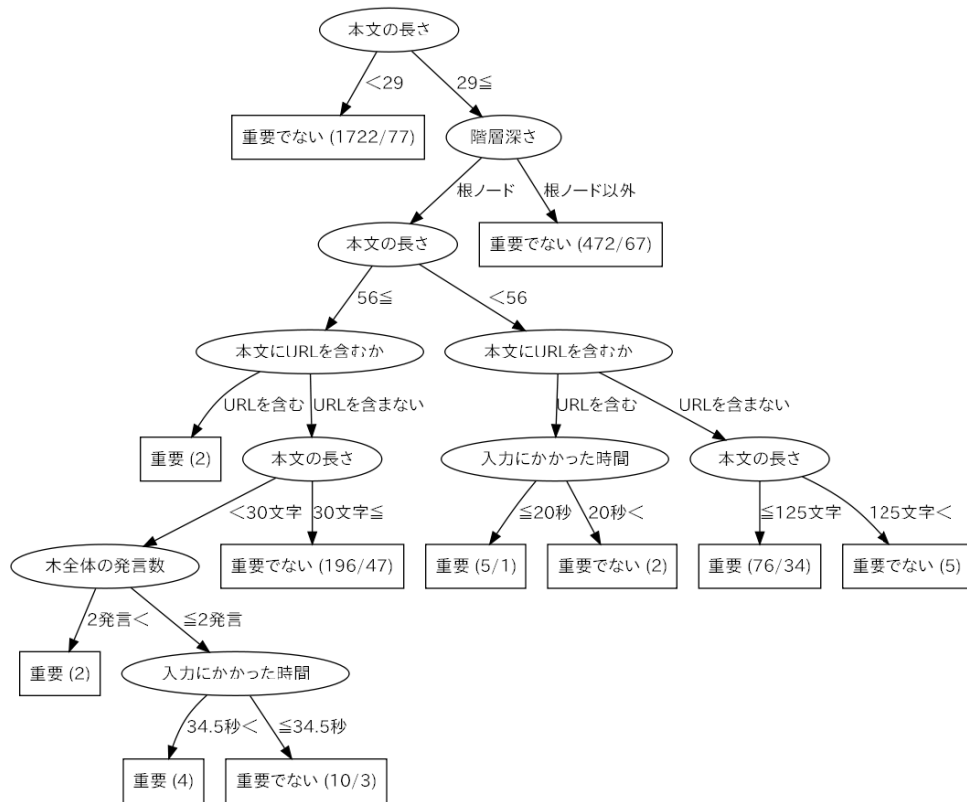


図 10 決定木クロスチャネル返信情報なし

Fig. 10 Result decision tree without XCRT parameter.

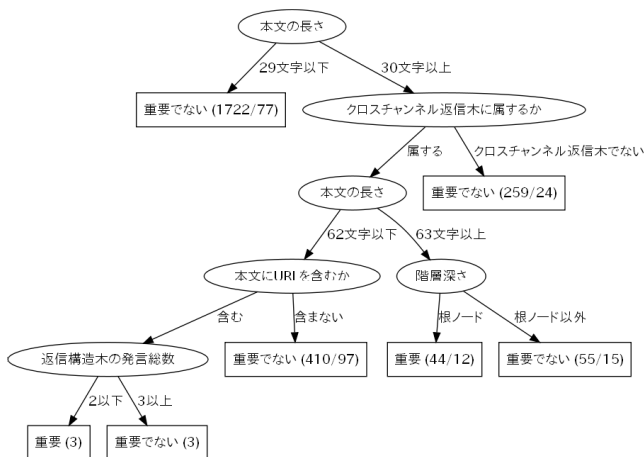


図 9 決定木クロスチャネル返信情報あり

Fig. 9 Result decision tree with XCRT parameter.

だからである。Full-XC 条件の決定木を図 9 に、Slide-XC 条件および No-XC 条件の決定木を図 10 に、それぞれ示す。図中の括弧は条件に合致したデータの数を表し、誤って分類されたデータがあれば「/」の後ろにそのデータ数が表される。Slide-XC 条件と No-XC 条件では、複雑な決定木が作成されたにもかかわらず適合率も再現率も低く、一定の法則が見いだせなかったため過剰な最適化が行われたと考えられる。

図 9 を見ると、Full-XC 条件のクロスチャネル返信に関するパラメータとしては、「クロスチャネル返信ではない」

のみが使用され、対口頭対話返信かどうか、および対スライド返信かどうかは区別されなかった。これは、クロスチャネル返信であれば対口頭対話返信であっても対スライド返信であっても重要であるかどうかの判定条件に違いはないということである。

Slide-XC 条件ではクロスチャネル返信に関わる情報として、対スライド返信であるかどうかのパラメータを与えているが、このパラメータは学習の結果出力された決定木では使用されなかった。これは、クロスチャネル返信木に属する発言の大半 (1,503 発言中 1,336 発言) が対口頭対話返信であり、対スライド返信が少ない (167 発言) ことによると思われる。以上の結果は、クロスチャネル返信の中でも、特に本研究で新たに提案した「対口頭対話返信」の有効性を示すものであるといえよう。

7. 考察

J4.8 ではクロスチャネル返信木であるかどうかという、たった 1 つのパラメータを取り除いただけで、適合率が 61.5% から 28.6% へと半分以下になってしまった。さらに、対スライド発言だけをクロスチャネル返信と見なす場合には性能は向上しなかった。ここからクロスチャネル返信木であるかどうか、とりわけ対口頭対話返信を扱うことが、推定性能に大きな影響を与えることが分かる。

ただし再現率はクロスチャネル返信木かどうかの情報を

用いた場合でもかなり低い結果となった。使用しているデータでは、発表1件あたり平均して124.8発言のチャット発言があり重要発言占有率が9.9%であることから、チャット・ログには平均して12.4発言の真の重要発言が含まれていると推測される。クロスチャネル返信木であるかどうかのパラメータを用いた場合は、再現率が12.9%、適合率が61.5%であるので、2.6発言の重要発言を推定し、うち1.6発言が真の重要発言である。

意見・質疑時間では時間的制約が重要な要素であり、限られた時間で優れた意見を得られるべきである。したがって、システムが重要でない発言を重要であると判定する (false-positive) 状態を可能な限り排除すべきであり、再現率が低くても適合率が高い状態が望ましいと考えられる。しかし、あまりに再現率が低ければそもそもチャットから1つも発言が推薦されないということであり、システムの存在意義がない。本来であれば、具体的に「何個」という形で発言の数を仮定する、たとえば「10分の意見質疑で5個の発言があればよい」と設定することは適切ではないが、妥当性を検討する参考としてあえて筆者らの経験を述べるとすれば、学会発表における質疑応答時間は5分から10分程度であり、質疑の数も5・6個程度である。チャット併用会議では質疑応答は口頭とチャットの両方から行われるため、チャットから2.6発言重要発言を推定しうち1.6発言が重要発言であったとしても実用に耐えらると思われ、実用的な精度に達しているといえる。

また、重要な発言を推定すること自体は、質疑応答の際だけではなく別のタイミングで使うことも考えられる。たとえば、発表後に自身の研究に対して得られた意見を反映させる場合などである。こうした場合では時間的制約がないため、false-positive をある程度許容し、見過ごしなどにより重要発言を取りこぼしてしまうことを回避する方が適切であると考えられる。こうした応用にも対応するため、再現率を高める手法について検討する必要もあるだろう。

8. まとめ

チャット上 (バックチャネル) から対面口頭対話上 (フロントチャネル) の内容に対しての明示的な返信を可能にするため、本論文で「クロスチャネル返信」を提案した。クロスチャネル返信機能を実装した Chatplexer を制作し、研究室のゼミで2カ月にわたって進捗報告のプレゼン発表とチャットのデータを収集した。収集したデータから、過半数の返信木はクロスチャネル返信木であると同時に、クロスチャネル返信木に属する発言の数も過半数を占めていること、さらに発表者にとって重要な発言はクロスチャネル返信木に8割以上含まれていることから、「発表者にとって重要な発言」の判定条件として、クロスチャネル返信であるかという情報を用いることの可能性が示唆された。

そこで、莫大なチャット・ログから「発表者にとって重

要な発言」だけを推定して抽出するシステムへの応用として、クロスチャネル返信木の情報を用いると重要発言推定にどのような影響を与えるかについて検証したところ、クロスチャネル返信木であるかという情報を用いると重要発言推定における適合率が28.6%から61.5%へと大きく改善されるということが分かった。既存のシステムですでに扱われているとおり、対スライド返信のみをクロスチャネル返信として扱う場合では、クロスチャネル返信情報は重要発言推定において重要な役割を果たさないことも分かった。さらに複数の言語的パラメータと組み合わせることにより、実用的なレベルで重要発言の推定システムを作ることが可能であると考えられる。

今後は日本語環境に限定し、重要発言がどのような言語的特徴を持っているかを検討することで重要発言推定システムを実装し、発表者とシステムが相互に関わりあう中でより良いチャット併用会議ができるかどうかを検討したい。

謝辞 本研究の一部は、平成21年度(財)栢森情報科学振興財団の研究助成を受けて実施された。ここに謝意を表す。

参考文献

- [1] Rekimoto, J., Ayatsuka, Y., Uoi, H. and Arai, T.: Adding another communication channel to reality: An experience with a chat-augmented conference, *CHI '98: Conference Summary on Human Factors in Computing Systems*, pp.271-272, ACM, New York, NY, USA (1998).
- [2] 綾塚祐二, 河口信夫: 参加者が作る会議支援システム—WISS Challenge, *コンピュータソフトウェア*, Vol.23, No.4, pp.76-81 (2006).
- [3] Golub, E.: On audience activities during presentations, *J. Comput. Small Coll.*, Vol.20, No.3, pp.38-46 (2005).
- [4] 平光節子, 白井正博, 杉山岳弘: チャットをベースにした会議のコミュニケーション活性化システムの検討, *情報処理学会研究報告, HI, ヒューマンインタフェース研究会報告*, Vol.2003, No.94, pp.7-12 (2003).
- [5] Hembrooke, H. and Gay, G.: The Laptop and the Lecture: The effects of multitasking in learning environments, *Journal of Computing in Higher Education*, Vol.15, No.1, pp.46-64 (2003).
- [6] 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, グループウェア*, Vol.2000, No.97, pp.37-42 (2000).
- [7] Harry, D., Green, J. and Donath, J.: backchan.nl: integrating backchannels in physical space, *CHI '09: Proc. 27th International Conference on Human Factors in Computing Systems*, pp.1361-1370, ACM, New York, NY, USA (2009).
- [8] 西田健志, 栗原一貴, 後藤真孝: On-Air Forum: リアルタイムコンテンツ視聴中のコミュニケーション支援システム, *WISS2009 第17回インタラクティブシステムとソフトウェアに関するワークショップ論文集*, pp.59-100 (2009).
- [9] 平島大志郎, 峯木寛明, 田中 充, 勅使河原可海: 会議録としての連続メディア情報の重要度を用いた検索方式の比較検討, *分散協調とモバイル (DICOMO2003) 予稿*

集, pp.353-356 (2003).

- [10] 由井蘭隆也, 重信智宏, 榎野晶文, 宗森 純: リアルタイムなコミュニケーション行為であるチャットへの意味タグ付加と電子ゼミナールへの適用, 情報処理学会論文誌, Vol.47, pp.161-171 (2006).
- [11] McCarthy, J.F. and Boyd, D.M.: Digital backchannels in shared physical spaces: experiences at an academic conference, *CHI '05: Extended Abstracts on Human Factors in Computing Systems*, pp.1641-1644, ACM, New York, NY, USA (2005).



小林 智也 (学生会員)

2004年静岡理工科大学機械工学科卒業。2006年北陸先端科学技術大学院大学知識科学研究科博士前期課程修了。現在、同大学院大学博士後期課程在学中。テキストチャットの応用システムに興味がある。



西本 一志 (正会員)

1987年京都大学大学院工学研究科機械工学専攻博士前期課程修了。同年松下電器産業(株)入社。1992年(株)ATR通信システム研究所研究員。1995年(株)ATR知能映像通信研究所客員研究員。1999年より北陸先端科学技術大学院大学助教授, 2007年より教授。2000~2003年科学技術振興事業団さきがけ研究21「情報と知」領域研究員兼任。1999年度情報処理学会坂井記念特別賞, 1999年度人工知能学会論文賞, ACM Multimedia 2004 Best Paper Award, 2010年度情報処理学会学会活動貢献賞, ICOST 2011 Best Multi-disciplinary Paper Awardほか受賞。IEEE computer society, ACM, 人工知能学会各会員。博士(工学)。