

議論内容の獲得と再利用に基づく 知識活動支援システムに関する研究

土田 貴裕

目次

概要	1
第1章 序論	5
1.1 背景	5
1.2 本研究の目的	8
1.3 問題点とアプローチ	8
1.4 本論文の構成	12
第2章 知識活動支援システムと知識活動サイクル	15
2.1 知識活動とは	15
2.2 知識活動支援システム	17
2.2.1 知識活動におけるコンテンツの再利用	18
2.2.2 知識活動マップ	19
2.2.3 知識活動支援システムの技術的要件	20
2.2.4 システム構成	21
2.2.5 コンテンツサーバ	22
2.2.6 SyncNode	25
2.2.7 RecNode	28
2.3 知識活動における会議と知識活動サイクル	30
2.3.1 知識活動における議論内容の再利用とその意義	30
2.3.2 DRIP サイクル	31
2.3.3 ディスカッションマイニングによる DRIP システムの構築	35
2.4 本章のまとめ	36
第3章 会議コンテンツの作成	37
3.1 システム構成	37
3.2 発表情報の取得	39
3.3 発言情報の取得	39
3.4 議論構造の取得	40
3.5 会議に関するメタデータの取得	42
3.6 会議コンテンツの妥当性に関する評価	42
3.6.1 議論構造の妥当性の評価	43

3.6.2	発言テキストの情報量に関する評価	47
3.7	運用実績とユーザビリティに関する考察	53
3.8	本章のまとめ	53
第4章	会議コンテンツの会議後および会議中における利用	55
4.1	会議コンテンツの検索・閲覧	55
4.1.1	インタフェース XML の作成	55
4.1.2	システム構成	58
4.1.3	インタフェース	59
4.1.4	会議コンテンツの利用状況に関する調査	66
4.2	会議コンテンツの利用に基づく議論の円滑化	71
4.2.1	議論中における会議コンテンツの可視化	73
4.2.2	過去の会議コンテンツの回顧	75
4.2.3	ユーザビリティに関する考察	78
4.3	本章のまとめ	79
第5章	会議コンテンツの再利用による知識活動支援	81
5.1	DRIP システムにおける知識活動マップの作成とその利用	81
5.1.1	知識活動マップの作成	81
5.1.2	知識活動マップのもたらす効果	84
5.2	議論内容の整理とタスクの決定	85
5.2.1	作成された会議コンテンツの取り込み	86
5.2.2	マーキングの付与	86
5.2.3	議論セグメントの取り込み	87
5.2.4	議論のまとめの記述・キーワードタグの付与	88
5.2.5	TODO の作成	89
5.3	整理した議論内容に基づくタスクの遂行	89
5.3.1	ノートの作成・編集・蓄積	89
5.3.2	コンテンツ間の参照情報の取得	90
5.3.3	ノートと TODO との関連付け	91
5.4	蓄積した情報を用いた発表資料の作成支援	91
5.5	評価実験	93
5.5.1	実験環境	93
5.5.2	評価手法	94
5.5.3	実験結果と考察	95
5.6	本章のまとめ	99

第 6 章	関連研究	101
6.1	議論内容の記録に関する研究	101
6.1.1	LiteMinutes	101
6.1.2	Portable Meeting Recorder	101
6.1.3	MinuteAid	102
6.2	議論の構造化・可視化に関する研究	103
6.2.1	gIBIS	103
6.2.2	Discussion Structure Visualizer	103
6.3	会議コンテンツの再利用に関する研究	104
第 7 章	結論	107
7.1	本論文のまとめ	107
7.2	今後の課題	109
7.3	将来の展望	109
7.3.1	知識活動支援システムの拡張	110
7.3.2	知識活動マップを用いた応用	112
7.3.3	グループの知識活動サイクルの支援	113
	謝辞	115
	参考文献	117
	研究業績	121

目 次

1.1	Discussion Recorder を利用しているゼミの風景	10
2.1	SECI モデル	16
2.2	組織的知識創造プロセスと研究活動	17
2.3	知識活動マップ	20
2.4	知識活動支援システムの構成図	22
2.5	URI を用いたコンテンツへのアクセス	23
2.6	プレゼンテーションサーバ	25
2.7	知識活動マップの簡略化	27
2.8	RecNode におけるコンテンツ・エレメントのインポート	29
2.9	DRIP サイクル	33
2.10	DRIP システムの構成	35
3.1	Discussion Recorder を利用しているゼミの風景	38
3.2	ディスカッションルーム	39
3.3	発表者用インタフェース	40
3.4	ポインタリモコン	41
3.5	書記用インタフェース	42
3.6	会議コンテンツ	43
3.7	複数参加者による図形の描画	44
3.8	システムが生成した議論構造と正解データとの比較	45
3.9	書記テキスト, 書き起こしテキストにおける自立語, キーワード, 共通キーワードの包含関係	50
3.10	書記による発言テキストの記述量の個人差	51
3.11	書き起こし率と再現率の相関関係	52
4.1	インタフェース XML	56
4.2	Discussion Browser のシステム構成図	59
4.3	Discussion Browser	60
4.4	ビデオビュー	61
4.5	層状シークバー	62
4.6	議論セグメントの概要表示	62

4.7	議事録ビュー	63
4.8	グラフビュー	64
4.9	検索の例	65
4.10	アンケート調査に用いた議論の判定木	67
4.11	アンケート調査に用いたアンケート用紙の一部	68
4.12	内容を覚えている議論セグメント数の推移	70
4.13	有効に利用された議論セグメント数の推移	70
4.14	被験者ごとの有効に利用された議論セグメント数の推移	71
4.15	所属年数による年間発言数の変化	72
4.16	所属年数による賛成発言およびマーキング発言の数の変化	72
4.17	Discussion Visualizer	74
4.18	議論セグメント内のシーケンスとその抽出手順	74
4.19	議論セグメントビューにおける親発言の修正	75
4.20	発言対象となるスライドの選択	76
4.21	Discussion Reminder	77
5.1	アノテーションの概念図	83
5.2	記録・蓄積された知識活動マップの可視化	84
5.3	リフレクションによって生成された知識活動マップ	86
5.4	会議コンテンツの追加を知らせるメッセージ	86
5.5	Discussion Browser 上でのマーキング	87
5.6	Discussion Browser と Discussion Mediator との連携	88
5.7	スライドサムネイルや発言情報を引用したまとめノート	89
5.8	TODO の作成インタフェース	90
5.9	ノート編集インタフェース	91
5.10	スライド作成インタフェース	92
5.11	DRIP システムの有無による発表内容に関するアンケート結果の差異	96
5.12	全発言/要対応発言に対するマーキング数の比較	96
5.13	Discussion Browser における会議コンテンツの閲覧回数	97
5.14	実験期間中に DRIP システムを用いて記録されたコンテンツ間の関係	98
5.15	引用によって作成されたノートから次のゼミへのつながり	99
7.1	ディスカッションブラウザにおける会議コンテンツ間の関連性の可視化	113

表 目 次

3.1	システムが生成した議論構造と正解データとの比較結果	46
3.2	書記テキスト, 書き起こしテキストにおける自立語, キーワード, 共通キーワードの数とその割合	49
4.1	被験者の回答から得られた議論セグメントの判定結果	69
5.1	実験で行われたゼミの時間 (分)	95
5.2	会議に行われた発言数および議論セグメント数 (括弧内は議論セグメント数)	95
5.3	発表者 B の未対応発言に対するマーキング	97

概要

企業におけるプロジェクトや大学研究室の研究活動のように、ある特定のテーマに関するアイデアを継続的に創造し、具体化・理論化する知識活動が広く行われている。知識活動には、議論や文献調査、システムの実装、実験・検証といったタスクが存在し、それぞれのタスクが相互に関係しながら繰り返し実行されることによって、新しいアイデアや知識が創造される。タスクを通じて生まれたアイデアや知識は、ノートやプレゼンテーション資料などのコンテンツとして記録されるが、コンテンツの作成・編集時に存在するコンテンツ間の関係情報は記録されておらず、成果物に至るまでの文脈情報が失われてしまう。このような文脈情報を獲得・利用することができれば、相互に関係しながら実行されるさまざまなタスクを統合的にサポートできるだろう。そこで本研究では、さまざまなコンテンツ間の引用情報を文脈情報として蓄積し、その文脈情報を用いて知識活動を統合的に支援するためのシステムである知識活動支援システムを実現した。

知識活動に内在するタスクの一つに会議がある。日々の知識活動を通じて蓄積してきたアイデアや知識を他者に向けて発表し、議論を行うことによって、意見やアドバイスなどのフィードバックを獲得できるという点で、会議は知識活動において重要な役割を果たしており、その再利用性も高い。そこで本論文では、会議を題材にして DRIP システムと呼ばれるシステムを知識活動支援システム上に実現した。

DRIP システムは、議論内容を記録した会議コンテンツの作成から再利用まで統合的にサポートすることで、知識活動の活性化を目指している。会議コンテンツの再利用とは、会議コンテンツの部分要素を引用することで、議論内容を自分なりの解釈で整理し、そこから生まれた新たなアイデアや知識との関連付けを行い、次の発表に用いる資料に取り込むことができるようにすることである。システムが会議コンテンツの再利用を支援することによって、過去の議論がどれだけ知識活動に活用されたかを機械的に蓄積できる。そのような情報を活用することで、システム利用者は、自分が今やるべきことを整理したり、過去の議論を踏まえた発表を行ったりすることができるようになる。

そこでまず、知識活動と繰り返し行われる議論との関係について考察することで、DRIP サイクルと呼ばれる、議論を中心とする以下の 4 つのフェーズから構成される知識活動のサイクルを提案する。DRIP システムは、DRIP サイクルの各フェーズを支援するためのさまざまなアプリケーション群から構成されている。

議論 (Discussion) フェーズ アイディアを他者に発表し，議論を行うことによって，多角的な視点による意見やアドバイスを獲得する

再認 (Reflection) フェーズ 議論の内容を整理し，その後に行うべきタスクを決定する

探求 (Investigation) フェーズ 議論内容を踏まえつつ，文献調査や実験・検証といったさまざまなタスクを遂行する

集約 (Preparation) フェーズ さまざまなタスクを通じて蓄積されたアイディアや知識をまとめ，次の議論のための発表資料を作成する

まず，議論フェーズでは会議コンテンツの作成を行うための Discussion Recorder，会議コンテンツの検索・閲覧を行うための Discussion Browser，議論中に会議コンテンツの可視化や回顧を行うことによって，議論の円滑化・活発化を支援するための Discussion Visualizer や Discussion Reminder を提案する．

Discussion Recorder は，議論の様子を映像・音声に記録し，発言テキストやスライドファイルの他に発言者情報や発言時間といったメタデータによって，議論の意味構造化を行う．その特徴として，会議参加者が専用のデバイスを使用することで，音声処理や画像処理，自然言語処理では取得が困難な，発言テキストや議論に関するメタデータなどの取得を行っている点が挙げられる．そこで，Discussion Recorder によって作成・蓄積された会議コンテンツを分析・評価することで，人間によって入力された発言テキストやメタデータが妥当であることを示す．

そして，メタデータの俯瞰やコンテンツの内部検索が可能な Discussion Browser によって，作成された会議コンテンツを必要ときに検索・閲覧し，過去の議論内容を参照できる．しかし，会議コンテンツが検索・閲覧できるようになるだけでは，議論内容を踏まえた知識活動を行うことが困難であることを示す．

次に，会議コンテンツを作成するだけでなく，それを利用することでリアルタイムに行われている議論を円滑に行うための試みについて述べる．Discussion Visualizer は，Discussion Recorder で取得しているメタデータをリアルタイムに可視化することによって，現在進行している議論の状態を確認するためのアプリケーションである．そして，Discussion Reminder は，過去の会議コンテンツを検索・回顧することで，参加者間の知識レベルの差異をなくし，議論をより活発なものにするためのアプリケーションである．本論文では，これらのアプリケーションの運用を通じて得られた考察についても述べる．

議論フェーズで用いるこれらのアプリケーションは，複数の人間が一堂に会して会議を行うため，共有スペースで使用されることを想定している．それに対して，再認フェーズや探求フェーズ，集約フェーズでは，システム利用者が個人用のスペースで作業するため，Discussion Mediator と呼ばれるクライアントアプリケーションを提供する．

再認フェーズでは、作成された会議コンテンツ内の発言を引用しながら、議論内容をまとめることができる。これにより、タスクの遂行中に過去の議論内容への参照が容易になる。また、他人の発言内容を自分の解釈で整理することによって、議論内容に対する理解を深めることができると考えられる。探求フェーズでは、再認フェーズを通じて生まれたタスクを遂行しながら新たなアイデアや知識を蓄積していく。Discussion Mediator は、アイデアや知識を記録する際に利用された会議コンテンツなどのコンテンツとの間に参照情報を自動的に生成する。これによって、会議コンテンツに基づいて創造されるアイデアや知識は、複数の会議コンテンツにわたり系統立てて記録・蓄積され、議論以外の知識活動と会議コンテンツとを密接につなぐと同時に、それらの検索や閲覧を容易にすることで知識活動の効率を上げることができる。そして、集約フェーズでは、それまでに蓄積された情報を活用することで、過去の議論を踏まえた発表資料の作成を行うことができる。

DRIP システムを用いて DRIP サイクルを繰り返し行うことによって、システムを利用している人間の知識活動が、過去の議論を適切に反映した、より密度の濃い効率的なものになると考えている。

第1章 序論

1.1 背景

近年，IT 革命という言葉に代表されるようにコンピュータやインターネットなどの情報技術の急速な発達に伴い，労働集約的な生産活動よりも知識集約的な生産活動が重視されるようになってきた．このような社会では，個人の持つ知識を組織全体の知識として共有・再利用し，新たな知識を効率的に創造できる仕組みが必要になる．

個人の持つ知識を組織的に共有し，組織内の知識創造を促すための手法としてナレッジマネジメントがある．1990 年代前半に野中らは，形式的言語では表現できない主観的・経験的な知識である暗黙知と，形式的言語で表現可能な客観的・理性的な知識である形式知が，個人・集団・組織の中で絶え間なく相互に作用しあうことによって知識が創造されるプロセスを SECI モデルと名付けて提言している [14]．SECI モデルにおいて知識創造のプロセスは，共同体験などによって暗黙知を獲得・伝達する共同化 (Socialization)，得られた暗黙知を共有できるように形式知に変換する表出化 (Externalization)，形式知同士を組み合わせる新たな形式知を創造する連結化 (Combination)，利用可能となった形式知を基に個人が実践を行い，その知識を体得する内面化 (Internalization) の 4 つのフェーズから構成される．そして，それぞれのフェーズが，ナレッジワーカー同士の暗黙知を共有するための場を通じて，個人・グループ・組織の相互作用の中で，ダイナミックに循環することによって個人および組織の知識が量的にも質的にも増幅されると主張している．これ以降，SECI モデルにおける各フェーズを（個別に，もしくは複合的に）支援するシステムであるナレッジマネジメントシステムに関する研究が数多く行われている．

SECI モデルが提言された当時は，特にナレッジワーカーが暗黙的に持っていた知識を，形式的言語を用いて記録する表出化を支援するためのシステムが生み出された．形式知として記録されていた知識は，顧客からのクレーム情報や，システム開発のノウハウ，受注事例などが中心であり，専門分野に関する知識の格納・検索に特化されたデータベース（知識ベース [18]）に関する技術が利用されていた．知識ベースは，ナレッジワーカーによって作成，編集，共有，検索されるものであり，このようなナレッジマネジメントシステムは第 1 世代と呼ばれる [31, 22]．

しかし，これらのシステムにはいくつかの問題が指摘されている．島津らは，ナ

レッジマネジメント活動を継続するうえでの課題として、「企業文化・風土や従業員の知識提供へのモチベーションの維持」を指摘している [31]。たとえば、「知識提供の貢献が業績評価への反映に対して小さかったり、本業とは異なる余分な仕事であったりすると、知識を提供する従業員のモチベーション維持が難しく」なったり、「質の高い知識がタイムリーに追加更新されないと、KM（ナレッジマネジメント）システムを利用する従業員の数が減り、それが提供者側の提供意欲への悪影響」につながると述べている。Davenport は、ナレッジワーカには情報を探したり配信したりする時間的な余裕がないため、業務に知識を埋め込んだシステムの必要性を説いている [5]。また、小林は、これまでのナレッジマネジメントシステムで蓄積される情報は設計書や提案書など業務に関する成果物だけであり、「どのようにしてその成果物が作成されたのか」という成果物に至るまでの文脈情報が蓄積されていないために、組織内で培われた知識がメンバー間で効率的に継承されない点を指摘している [27]。

これに対して、第2世代と呼ばれるナレッジマネジメントシステム [22] では、ナレッジワーカが意識的に定義する形式知だけでなく、ナレッジワーカが人事システムやスケジュール管理システムなどの業務アプリケーションを使って業務をすることで生産される情報が注目された。また、これに並行するように、情報処理技術でナレッジワーカ間の協調作業を支援する取り組みが行われるようになった。なかでも一般的な技術であるグループウェアは、CSCW (Computer Supported Cooperative Work) と呼ばれるコンピュータによる協同作業支援に関するシステムの総称であり、具体的なものとしてはサイボウズ¹や Lotus Notes/Domino²などが挙げられる。これらのシステムには、組織内部や外部とのコミュニケーションを円滑に行うための Web メール機能、メンバーとスケジュールを共有するスケジュール機能、メンバー間の打ち合わせや議論を行うための掲示板機能など組織内の知識創造を支援するさまざまな機能が存在する。グループウェアの有用性は徐々に認識されており、民間企業の半分以上がグループウェアを導入しているという調査結果も報告されている³。

さらには、ナレッジワーカに報告書や提案書といった成果物を直接登録させる形態ではなく、ナレッジワーカの活動内容を記録し、マイニングする形態のナレッジマネジメントシステムも注目されている [31]。このようなシステムは、より個人の活動に近いメールや社内掲示板・ブログなどのシステムを利用したコミュニケーションから知識を抽出しようという試みである。具体的な製品として、Knowledge Market⁴などがあり、ナレッジワーカの活動を中心に据え、ナレッジマネジメント

¹サイボウズ株式会社：サイボウズ。 <http://cybozu.co.jp/> (2010 年 12 月 11 日)

²日本アイ・ビー・エム株式会社：Lotus Notes/Domino。
<http://www-06.ibm.com/software/jp/lotus/products/nd85/> (2010 年 12 月 11 日)

³株式会社ノークリサーチ：2008 年中堅・中小企業の IT 投資動向に関する調査報告。
<http://www.norkresearch.co.jp/pdf/2008lait.pdf> (2010 年 12 月 11 日)

⁴リアルコム株式会社：ナレッジマネジメントツール「KnowledgeMarket EnterpriseSuite」。

を業務プロセスに埋め込むことで、知識を効率的に抽出し、利用できるようにすることに注力したシステムを実現しようとしている。

このように、ナレッジマネジメントシステムに関する研究・開発は幅広く行われているが、著者は、このようなナレッジマネジメントシステムには以下の問題点がある。

- 自然言語だけでは表現しきれない活動内容の記録が十分に行われていない
- 成果物に至る文脈情報の記録が十分に行われていない。また、取得した文脈情報に基づく活動支援が十分に行われていない

自然言語だけでは表現しきれない活動内容の記録 これまでに触れてきたナレッジマネジメントシステムの対象は、メールや掲示板・ブログのように自然言語で記述された情報が中心である。しかし、組織におけるナレッジワーカーの活動には、自然言語だけでは表現できないような情報を含むものも存在する。

角らは、インターネットと実世界を対比して、インターネットは言語に大きく依存した情報がやり取りされているが、実世界の社会生活における対面コミュニケーションは言語情報だけでなく、身振り手振り、視線、立ち位置や姿勢、表情といった多くの非言語的な情報に支えられていると述べている [32]。このような、人間同士が実際に対面してさまざまな状況を共有しながら行われる、言語的情報だけでなく非言語的情報を含む実世界インタラクションでは、より多くの暗黙知がやり取りされている。しかしながら、多くの実世界インタラクションは、記録が難しいために形式化されず、ナレッジマネジメントシステムに取り込むことができていない。

成果物に至る文脈情報の記録およびその利用 成果物に至る文脈情報を記録することで、その成果物のきっかけとなる現象や要求、議論を通じて得られた結論や問題点などを芋づる式に得ることができる。メールや掲示板のメッセージ間の返信関係も文脈情報の一つであり、返信関係をスレッド表示することで議論の流れを追いやくなる。

そして、前述した実世界インタラクションに関する記録を文脈情報に取り込むことでその価値はさらに高まる。メールでやり取りされる議事録には、最終的な結論や決定事項だけが記載されていることが多いが、場合によっては議事録を作成する際に欠落した情報が必要となる場面も存在する。たとえば、決定事項に至るまでにどのような問題点が指摘されたのか、その解決策に対してどのような意見が出たのか、といった情報は必ずしも議事録に記載されているとは限らない。そこで、会議の様子を記録した映像・音声情報と決定事項をリンクすることで、詳細な内容を確認できる。しかし、これまでのナレッジマネジメントシステムには、

<http://www.realcom.co.jp/it/knowledgemarket.html> (2010 年 12 月 11 日)

実世界インタラクションの内容を記録する仕組み自体が十分ではないため、それを文脈情報に含めることはなおさら困難である。

1.2 本研究の目的

本研究の目的は、上記の問題点を解決するシステムの実現である。なお本研究では、ある特定のテーマに関するアイデアを継続的に創造し、具体化・理論化する活動を知識活動と呼び、知識活動に関する問題を解決するためのシステムを知識活動支援システムと呼ぶ。知識活動支援システムは、メールやブログといった自然言語で表現できるような情報や実世界インタラクションの内容を記録した情報を、全てコンテンツとして扱い、コンテンツの部分要素間の関係情報を収集・蓄積・共有し、さまざまな応用を実現するシステムである。

特に、本論文では、実世界インタラクションの一つである会議を題材とし、会議を中心とする文脈情報の収集・蓄積、および知識活動支援に関する手法について述べる。会議にはさまざまな形態があり、たとえば、商品のアイデアを出し合うようなブレインストーミング、些細な話題について話し合うような限られた人数で行われる簡単なミーティングなどが挙げられる。本研究では、大学研究室におけるゼミのように、参加者が一堂に会して行われる発表形式の対面同期型の会議に着目する。このような会議では、最近のグループウェアが新たに解析の対象に加えている、メールや掲示板での非同期型コミュニケーションでは得られない、対面同期型コミュニケーションならではの、より自然で豊富な情報がやり取りされている。これらの情報を記録・構造化しコンテンツ化することで、形式化された知識同士の関係情報のような、これまでのナレッジマネジメントでは抽出できない情報を抽出できるだろう。

1.3 問題点とアプローチ

本論文では、著者が所属する研究室で研究・開発を行っているディスカッションマイニングと呼ばれる技術と、知識活動支援システムを組み合わせることによって、DRIP システムと呼ばれるシステムを実現する。DRIP システムは、大きく分けて以下に示す要件を統合的に扱うものである。

- 会議情報の獲得による会議コンテンツの作成
- 会議コンテンツを用いた議論の円滑化支援
- 会議間の活動における会議コンテンツの再利用

以下では、それぞれの技術に関する背景や問題点、その問題点へのアプローチについて述べる。

会議情報の獲得による会議コンテンツの作成 会議を記録し計算機で取り扱えるようにするための研究は CSCW の分野でも数多く行われている。一つは、映像・音声情報を解析し、参加者の挙動や音声の変化を抽出したものを利用する方法である [4, 10, 17]。これらの研究では、会議の様子を撮影した映像・音声情報を解析し、会議参加者の挙動や音量の変化をイベントとして抽出することで、動画を閲覧する際の手がかりとして利用している。また、会議の議事録を音声認識により作成し、インデックス化を行う研究がある [30]。しかし、これらの手法では、パターン認識の精度に限界があるため正確な情報が得られず、現在のところあまり実用的であるとは言えない。さらに、参加者の挙動や音声の変化が必ずしも議論内容に合致したものではないため、議論構造や会話の文脈などの意味を考慮した解析を行うことが困難である。

それに対して、会議の議事録に必要な情報を人間が人手で入力する方法がある。たとえば、テープレコーダで録音して後から書き起こす方法や、速記のように議論内容を特別な符号を用いて書き取り、後から通常の文字に書き直す方法などが挙げられる。前者は会議後に入力を行う方法であり、後者は会議中に入力を行う方法である。前者は、何度も議論内容を確認できるので、より詳細な内容を獲得することができる。しかし、会議後に行われる入力作業は、その後の知識活動に密接にかかわる情報を、議事録の閲覧行為を通じて整理・抽出することに注力されるべきであり、明確な目標が見えにくい独立した入力作業は、そのモチベーションを維持することが著しく困難であるため、会議中にできるだけ自然に獲得できる方法を考える必要がある。ただし、会議ごとに速記のような特別なスキルを持つ人間を用意することは現実的ではないため、会議参加者が入力作業を行えることが望ましい。その場合、参加者が議論内容を把握しながら入力を行うため、パターン認識では取得が困難な意味情報を取得できる反面、会議参加者への負担が大きくなってしまい、場合によっては会議そのものを阻害する可能性もある。そのため、入力のための操作をできるだけ簡略化することで、会議参加者の負担を軽減する研究も行われている [23]。

そこで、本研究では、情報技術によって解決可能なことは情報技術で解決し、人間にしかできないと思われることはできるだけ効率的に行えるように支援するという思想に基づいて、Discussion Recorder と呼ばれる、会議情報を記録・構造化することで会議コンテンツの作成を行うアプリケーションを提案する。Discussion Recorder では、図 1.1 のように、会議参加者が一堂に会して行われる発表形式の会議を想定している。Discussion Recorder は、数台のカメラやマイクによって映像・音声情報を記録するだけでなく、会議参加者が専用のデバイスを操作することで、発言時間などのインデックス情報や発言間の関連などの意味情報を付与できる。

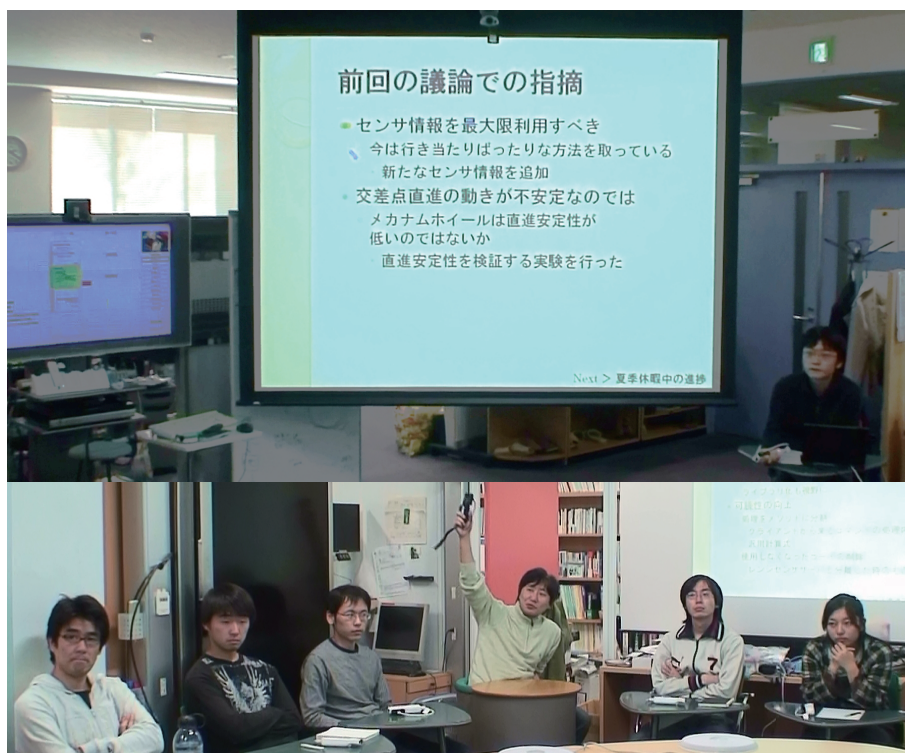


図 1.1: Discussion Recorder を利用しているゼミの風景

会議コンテンツを用いた議論の円滑化 近年では、会議はクリエイティブな活動であるという認識が広まっており、これまでに行われてきた形骸化された会議そのものを見直すことに注目が集まっている。これまでの会議は、目的が不明瞭である、立場の低いものが発言しにくい雰囲気があるなどさまざまな問題点が指摘されており [43]，これらの問題点を解消するためのノウハウを紹介する書籍などが多く出版されている。斎藤はこのようなノウハウを「十の法則」としてまとめ、これらの法則に基づき独自の会議スタイルである「三つの革命」を提案している [28]。

会議に関するノウハウの中に、議論内容を参加者間で共有しながら会議を進めることで議論を円滑にする手法が提案されている。eXtreme Meeting⁵は、「会議はそもそも議事録を共同で作成する作業である」という定義に基づき、会議参加者全員で議事録を共同で作成する Web ツールを提供している。また、議論の内容をホワイトボードなどに文字や図形を用いて表現することで、議論のプロセスの共有とメンバーの対等な参加を促進するための技法であるファシリテーション・グラフィック [40] も提案されている。

⁵株式会社サルガッソー：eXtreme Meeting。
<http://xm.sargasso.jp/xm.php> (2010 年 12 月 11 日)

ディスカッションマイニングでは、会議コンテンツに必要となるさまざまな情報を取得するだけでなく、取得した情報をその場で利用することで、現在行われている議論そのものを円滑化させる方法についても検討を行っている[34]。たとえば、取得している最中の情報を可視化することによって、発言者は現在行われている議論内容を簡単に把握でき、自身の発言内容を整理できるだろう。また、誰が過去のどの発言を受けて発言しようとしているのかを理解することで、その人に発言権を譲るなど、議論を効率的に進められるようなファシリテーションを参加者に促すこともできるだろう。そこで DRIP システムでは、議論内容の可視化を行うアプリケーションである Discussion Visualizer を提供する。

また、参加者全員が発表に関して同じ知識量を持ち合わせていないために、参加者全員が活発に議論できない状況が考えられる。そこで、現在行われている議論に関連のある、過去の会議コンテンツを検索・回顧することで、会議の参加者全員が議論に参加し発言を行い易くなり、結果として議論の活性化につながると考えた。DRIP システムでは、Discussion Reminder と呼ばれるアプリケーションを提供することで過去の会議コンテンツの検索・回顧を行う。

さらに、作成された会議コンテンツを検索・閲覧できる仕組みを提供することで、過去の議論内容を参照しながら議論後のタスクを遂行できる。そのため、Discussion Recorder によって取得したメタデータを俯瞰し、発言やスライドといった会議コンテンツの内部まで検索が可能なアプリケーションである Discussion Browser を実現した。

会議間の活動における会議コンテンツの再利用 大学研究室のゼミのように、発表者が日常の活動を通じて生み出したアイデアや成果を報告し、それに対して参加者が議論を行うような会議は、一度だけ行えば全ての問題が解決するわけではなく、繰り返し行われる。そこで重要になるのは、議論内容が発表者の会議後の活動内容に反映され、その活動を通じて生まれたアイデアや成果を再び会議の中で報告することでさらなる議論が生まれる点である。もし、議論内容が正しく発表者の活動内容に反映されていなければ、参加者が会議への参加や議論に費やした労力が無駄になったことになる。また、過去の議論内容を踏まえていない状態のまま、次の会議に臨んだとしても、冗長な議論や真に議論すべきことから逸脱した議論が多くなり、その後の活動が円滑に行われない可能性がある。

作成された会議コンテンツには、発表者の今後の活動にとって重要な意味を持つような議論もあれば、発表内容に対する単純な質疑応答のような創造性の低い議論も混在している。このような発言の重要度は、発言内容を参照する人によって異なり、機械的に推定することが困難なため、会議コンテンツの作成時に取得することができない。そのため、会議後に重要な発言内容を参照することが手間のかかる作業になることがある。そこで、発表者が会議コンテンツの中で自身にとって重要な発言を取捨選択し、それらを引用しながら自分なりの解釈を記述す

ることで議論内容を整理することが必要となる．議論内容を自分の中でしっかりと消化するためには，会議コンテンツをただ作成・検索・閲覧するのではなく，会議コンテンツを一度分解し，自身にとって必要な箇所を抜き出して，再構築することが必要である．本研究では，会議コンテンツを分解・再構築し，そこから新たなアイデアや知識を創出したり，以降の発表に用いる資料に取り込んだりすることを「会議コンテンツの再利用」と定義している．

ディスカッションマイニングシステムが会議コンテンツの再利用を支援することで，利用者はいつでも必要な情報を容易に参照でき，議論内容を適切に活動内容に反映できるようになる．そして，次に行われる会議において，過去の議論内容に基づく活動を通じて創造されたアイデアや知識を，過不足なく正確に会議参加者に伝達することで，さらなるフィードバックを得ることができる．

さらに，このディスカッションマイニングシステムによって蓄積された情報を用いて，さまざまな応用を実現できる．たとえば，議論と蓄積された新たなアイデアや知識を関連付けることが可能なため，関連付けられたアイデアや知識の量によって，自分がどの議論を集中的に扱っているのか，また逆にどの議論を疎かにしているのかを把握できる．当然ながら，量だけで議論と活動の密接度や活動の成熟度を測ることは難しいが，情報の更新時間や更新頻度などさまざまな情報を組み合わせることによって，より適切に評価できるようになるだろう．

また，蓄積された情報は利用している人間のライフログとして利用することもできる．たとえば，過去の自分の発表や発言の変遷をたどることで反省や発見を促したり，他者の活動履歴を閲覧したりすることでその人間にまつわる文脈情報の理解を促進することが可能になる．

さらに本論文で提案するディスカッションマイニングシステムを繰り返し利用することによって，重要発言の発見や生産性の高い活動を行うための方法論といった知識発見も実現できると考えている．Discussion Recorder では，発言テキストや発言に含まれるキーワードの数，発言者の情報，発言時間といったメタデータを用いた機械学習による重要発言の抽出も行ってきたが[36]，その精度はあまり高いものではなかった．しかし，ディスカッションマイニングシステムで蓄積された情報を用いることで，発表者の知識活動に影響を及ぼした議論を容易に抽出できるようになる．また，生産性が高い人間のデータと生産性があまり高くない人間のデータを比較・分析することで，生産性を向上させるために必要となるファクターを発見できるだろう．

1.4 本論文の構成

本論文の構成を以下に示す．

まず、2章では本研究で対象としている知識活動の定義について述べる。そして、知識活動を統合的に支援するためのシステムである知識活動支援システムについて述べる。次に、知識活動に内在するタスクの一つである会議に着目し、知識活動における議論内容の再利用の意義について述べる。そして、DRIP サイクルと呼ばれる知識活動サイクルを定義することで、知識活動を円滑に行うための方法について考察する。最後に、知識活動支援システムを基にして DRIP サイクルを統合的に支援する DRIP システムについて述べる。3章では、会議に関するさまざまな情報を獲得し、会議コンテンツを作成するための仕組みとして Discussion Recorder を提案する。Discussion Recorder で獲得している情報の種類や、その取得方法について述べる。Discussion Recorder の特徴は、発言の開始・終了時間のように現状の自動認識技術では認識が困難な情報を人間が入力することで、半自動的に会議コンテンツを作成するという点である。また、長期間にわたる運用を通じて蓄積された会議コンテンツを分析することで、会議コンテンツの妥当性を評価する。4章では、会議後および会議中における会議コンテンツの利用について述べる。会議後における利用として、会議コンテンツの検索・閲覧を行う Discussion Browser を提案する。また、会議中における利用として、取得している情報を可視化する Discussion Visualizer と過去の会議コンテンツの検索・回顧を行う Discussion Reminder を提案する。5章では、DRIP サイクルの中で会議コンテンツの再利用を支援する仕組みである DRIP システムについて述べる。また、DRIP システムを用いるユーザと用いないユーザとの比較実験を行い、DRIP システムの有効性を確認する。さらに6章では関連研究について述べ、7章で本論文をまとめ、今後の課題に加え将来の展望について述べる。

第2章 知識活動支援システムと知識活動サイクル

本章では、まず本研究で支援の対象にしている知識活動の定義、および知識活動支援システムの概略について述べる。知識活動支援システムそのものは、汎用的なシステムであり、独自の仕様に沿ったコンテンツ構造の設計（画像やビデオなど一般的なコンテンツに関してはあらかじめ定義されている）と、データ送受信のプロトコルの実装をすることでシステムの拡張が可能である。

本論文は、実世界インタラクションの一つである会議を題材にして知識活動支援システムの実現例を示すことを目的の一つとしている。それにあたって、本章では会議を中心に、知識活動支援システムを効果的に運用する際に重要となる、会議の再利用性、そして会議中心の知識活動サイクルについても述べる。このサイクルを DRIP サイクルと呼び、サイクル内の各フェーズと会議記録との関係性について考察する。

2.1 知識活動とは

知識創造企業やナレッジマネジメントなどの言葉に代表されるように、現代社会における知識の重要性が高まっている。今後、どのようにして質の高い知識をより多く創造・管理できるかが重要になると予想される。

野中らは、知識創造の枠組みには認識論的次元と存在論的次元の2つの次元があることを指摘している[14]。認識論的な次元では、知識には形式的言語で表現可能な形式知と表現不可能な暗黙知の2つがあり、これらの知識が相互作用することによって知識が創造される。この形式知と暗黙知の相互作用はSECIモデルと呼ばれ、以下の4つのプロセスのスパイラルとして表現される(図2.1)。

共同化 (Socialization) 経験を共有することによって、個人の暗黙知からグループの暗黙知を創造するプロセス

表出化 (Externalization) 暗黙知を、メタファ、コンセプト、モデルなどの共有できる形式知に変換するプロセス

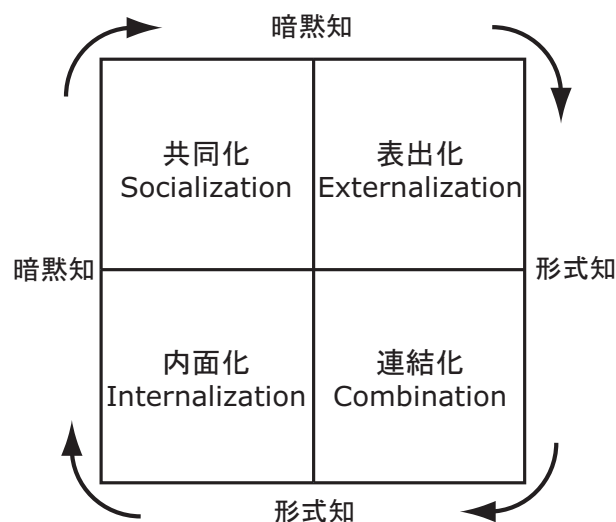


図 2.1: SECI モデル

連結化 (Combination) 共有された形式知を組み合わせる新たな知識体系を創り出すプロセス

内面化 (Internalization) 行動による学習によって形式知を暗黙知として体得するプロセス

また、存在論的な次元では、個人が創造した知識が組織的に拡張され、グループや組織内、そして組織間にわたる知識として結晶化される。この過程は共同化に相当する暗黙知の共有から始まる。そして、共有された暗黙知は新しいコンセプトという形の形式知へ変換される。そして、変換されたコンセプトが追及する価値があるかを示すために正当化を行った後に、プロトタイプなどの原型を作成する。このようなプロセスを通じて構築された知識はグループや組織、そして組織外へ移転される。このような 1) 暗黙知の共有、2) コンセプトの創造、3) コンセプトの正当化、4) 原型の構築、5) 知識の移転という 5 つのフェーズを経ることによって、スパイラルが生まれ、個人の知識が組織の知識として変換されていく。そして、2 つの次元における知識スパイラルが複合的に実践されることによって、組織的な知識創造が行われる。

野中らは、企業組織における知識創造を中心に述べていたが、視野を広げて捉えれば研究活動にもこの考え方を適用することができる (図 2.2)。研究活動では、新たな問題点や未解決の問題点に対するテーマを設定し、独自の視点から解決策を提案する。そして、実験・検証を通じて自身のテーマの妥当性を確認し、その成果を論文にまとめて公表する。論文を公表することによって、関連する分野の研究者の中で研究成果が共有され、新たな問題点が指摘されたり、新たなアプローチが提案されたりする。そして、このプロセスが繰り返し行われることによって、

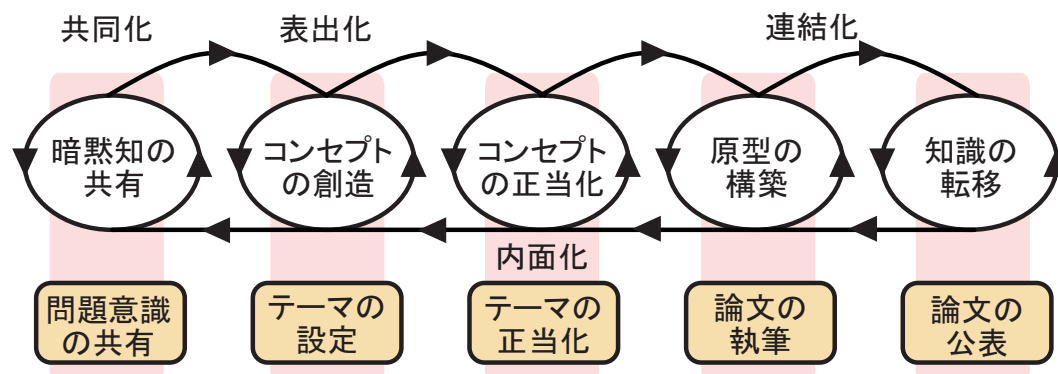


図 2.2: 組織的知識創造プロセスと研究活動

社会に貢献するような技術が生み出される。このように、企業組織における知識創造や研究活動に共通する点を踏まえることによって、より広範囲な知識創造支援を実現できると考えられる。

企業組織における知識創造や研究活動に共通する点は、テーマ（コンセプト）の一貫性である。たとえば、企業で新たな商品コンセプトが生み出されれば、企業内のメンバーがそれぞれの視点からそのコンセプトを満たすためのアイディアを出し合い、まとめていくことによって新しい商品が生み出される。また、研究活動では、いまだに解決されていない問題点を研究のテーマとして扱う。その問題点を解決するために独自の視点からアプローチを行い、新たな知識や技術を生み出していく。本研究では、企業組織における知識創造や研究活動のように「あるテーマに対して継続的にアイディアを創造し、知識として理論化・具体化する活動」を知識活動と呼び、情報技術を用いることによって知識活動を支援することを目指す。

また知識活動は、個人で行うものと複数人のグループで行うものの2種類が存在する。グループの知識活動も突き詰めれば、個人の知識活動の組み合わせと捉えられる。グループの知識活動には、テーマに対する認識の共有やメンバー間の進捗状況の把握など、個人の知識活動には存在しない要素が存在することは確かだが、グループを構成する個々人の知識活動を支援することで、結果としてグループの知識活動を支援できる。そのため、本研究では個人の知識活動に着目し、その支援を行う。

2.2 知識活動支援システム

前章では、ナレッジワーカー個人および組織における知識創造を支援するナレッジマネジメントシステムに関する次の問題点を指摘した。

- 自然言語だけでは表現しきれない活動内容の記録が十分に行われていない
- 成果物に至る文脈情報の記録が行われていない．また，取得した文脈情報に基づく活動支援が十分に行われていない

本研究では，これらの問題点を解決するため，知識活動支援システムを提案する．知識活動支援システムは，メールやブログといった文字列によって表現される情報や，会議のような実世界インタラクションに関する情報をすべてコンテンツとして扱い，コンテンツの再利用を通じてコンテンツの部分要素間のリンク情報を収集・蓄積・共有することで，知識活動を支援するための応用を実現するためのシステムである．

本節では，まず知識活動支援システムにおけるコンテンツそのもの，そしてコンテンツの再利用の定義について述べる．そして，知識活動支援システム全体の構成に触れたのち，コンテンツの再利用を実現するために作成した知識活動支援システムの機能について述べる．

2.2.1 知識活動におけるコンテンツの再利用

一般にコンテンツは，テレビ番組や映画，音楽のように多数の人間に対して公開されている娯楽用のメディアという印象がある．辞典によれば，コンテンツとは以下のような意味を持っている¹．

1. 中身．内容．
2. 書籍の目次．
3. 放送やインターネットで提供されるテキスト・音声・動画などの情報の内容．

最後の定義に従うと，1. それ自体が意味のある内容を持っている，2. (Web 上のコンテンツの場合) インターネットを通じて公開されている，という2点がコンテンツとして成立するための要件だと考えることができる．本研究では，このような要件を踏まえながらコンテンツの概念を再定義する．

知識活動では，成果報告書やプレゼンテーション資料のように，これまでの活動内容をまとめて他人に公表するための文書や，その日行わなければならないTODO，気付いたことを簡潔にまとめたメモのように日常的に多くのデータの作成・編集を行っている．成果報告書やプレゼンテーション資料はもちろんのこと，TODOやメモのように一見すると個人的なものだと思われるデータも状況に応じて他人と共有することがある．たとえば，大学研究室の先輩が後輩の提案した研究計画をチェックするために，個々のTODOの内容をチェックするような状況が挙げら

¹新村 出編，広辞苑 第六版，岩波出版，2008 年．

れる．そこで本研究では，知識活動を通じて生み出されるさまざまなデータをコンテンツと呼ぶことにする．

次にコンテンツの再利用の定義について述べるにあたって，われわれが日常的に行っているコンテンツ（の部分要素）の参照・引用について触れておく．われわれは，過去のプレゼンテーション資料の一部をコピーして新しいプレゼンテーション資料を作成したり，同じ画像ファイルを成果報告書や論文に使いまわしたりするように，日常的にコンテンツ（の部分要素）の参照・引用を行っている．このことをもっとも象徴しているのが，コピーアンドペーストと呼ばれるコンピュータ上の編集操作である．コピーアンドペーストは，テキストだけでなく，ファイルそのものや画像，文書のレイアウト情報などをクリップボードと呼ばれるデータ領域にコピーし，異なる箇所へその内容を複製する機能であり，コンテンツの引用を容易に行うことができる．

コピーアンドペーストの問題点は，使用時に引用元および引用先の関係情報が記録されないことである．結果として，時間が経過するにつれてペーストされた情報が，どこからコピーされたものなのかが分からなくなってしまい，成果物に至るまでの文脈情報が失われてしまう（そもそもそれがペーストされた情報なのかどうかも分からなくなる可能性もある）．そこで本研究では，引用元および引用先の関係情報を記録しながら，コピーアンドペーストのようにコンテンツの部分要素を引用しながら新たなコンテンツを生み出すことをコンテンツの再利用と呼ぶことにする．「利用」ではなく「再利用」という言葉を用いているのは，引用元の本来の目的（閲覧する，検索するなど）にとらわれず，さまざまなコンテンツにさまざまな用途を持って引用できる，という意味を含んでいるからである．

2.2.2 知識活動マップ

テーマに対する目標への達成度（達成できていない場合は，それに加えて現状及び問題点）を知ることによって，達成度に対する満足感からモチベーションを引き出したり，次に取り組むべきタスクを決定したりしやすくなるため，円滑な知識活動の遂行につながると思われる．そのため，知識活動支援システムでは，「どのようなタスクを行い，どのようなアイデアや知識を生み出したか」という知識活動に関する文脈情報の収集・蓄積を行う．この文脈情報を本研究では知識活動マップと呼ぶ．

知識活動マップは，図 2.3 のように，活動を通じて作成されるコンテンツもしくはコンテンツの部分要素（以下，エレメントと呼ぶ）をノード，エレメントの引用時に得られるリンク情報をエッジとするグラフ構造を持っている．テキスト文書内のようにコンテンツの中に含まれている部分要素を定義できるコンテンツを引用した場合は，その部分要素（章や段落，文章など）がノードとなりえるが，画像

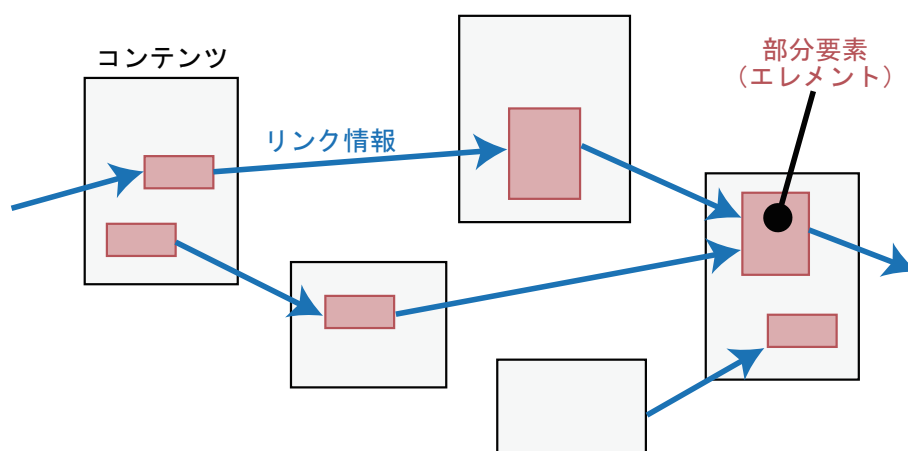


図 2.3: 知識活動マップ

のように部分要素を明確に定義できないようなコンテンツを引用した場合は、コンテンツそのものがノードとなる。

本論文では、個人が行う活動を対象としているため、本システムで作成される知識活動マップは、原則として個人に特化したものになるが、複数の人間の知識活動マップを統合して、グループの知識活動マップを作成することも可能である。

2.2.3 知識活動支援システムの技術的要件

知識活動支援システムによって知識活動マップを蓄積する際に解決すべき技術的要件には以下のようなものが挙げられる。

- 任意のコンテンツまたはエレメントを参照できるポインタ
- ポインタが参照しているコンテンツまたはエレメントにアクセスする仕組み
- コンテンツまたはエレメントのリンク情報を集約する仕組み

知識活動マップは、知識活動を通じて作成・編集される全てのコンテンツに対応していることが望ましい。しかし、コンテンツの種類によって、そのコンテンツの持ち得るエレメントの形式が異なるため、任意のコンテンツのエレメントを参照できるような柔軟な表現形式をもつポインタを用意し、そのポインタそのものを知識活動マップのノードとして用いることで対応できる。また、コンテンツそのものがノードとなる可能性もあるため（たとえば画像の引用）、そのポインタはコンテンツ自体も参照できるような表現形式である必要がある。

ポインタはあくまでコンテンツやエレメントを参照するための情報でしかなく、それ自体は意味情報を持っていない。つまり、ユーザが引用元の情報を参照する

際，ポインタが指示しているコンテンツやエレメントへアクセスし，その内容を確認できる仕組みを提供する必要がある．

ユーザは日常的な活動の中でエレメントの引用を行うため，ユーザが普段使用するようなツール（メモ帳やプレゼンテーションソフトなど）にその機能を持たせることが望ましい．つまり，引用機能を持ち合わせたクライアントアプリケーションを自由に作成し，その操作履歴から収集されたリンク情報を1つのサーバに集約させるような仕組みが必要となる．また，エレメントを引用するためには，必要とするエレメントを検索する機能が必要となる．

2.2.4 システム構成

本研究で提案する知識活動支援システムの構成を図2.4に示す．知識活動支援システムは，おもに以下に示すものから構成されている．

- コンテンツサーバ（Webサーバ）
- RecNode（クライアントアプリケーション）
- SyncNode（Webサーバ）

コンテンツサーバは，知識活動におけるコンテンツの作成・編集・公開を管理し，コンテンツの編集情報を SyncNode へ逐次送信する機能を持つ Web サーバである．利用するコンテンツの種類に応じて，異なるコンテンツサーバを用意する必要がある．本論文では，比較的良好に利用されるコンテンツである画像・プレゼンテーション資料・テキストファイルの3種類に対応するコンテンツサーバをあらかじめ提供している（それぞれイメージサーバ・プレゼンテーションサーバ・ドキュメントサーバと呼ぶ）．これら以外のコンテンツ（たとえば動画や論文）に対応するコンテンツサーバを実現する場合は，後述するコンテンツのデータ構造やデータ通信のプロトコルに関する仕様に準拠する必要がある．

SyncNode は，コンテンツサーバから送信されるコンテンツの編集情報や RecNode から送信されるエレメントの引用情報を知識活動マップとして蓄積し，必要に応じてその情報をコンテンツサーバや RecNode へ提供する機能を持つ Web サーバである．

RecNode は，ユーザが日常的に使用する機能（メモを書く，など）を提供し，その操作の中で行われたコンテンツの部分要素の引用に関するログ情報を収集し，SyncNode へ逐次送信するためのクライアントアプリケーションである．コンテンツの部分要素を引用する際に必要となるコンテンツ情報は，必要に応じてコンテンツサーバから取得する．また，SyncNode から知識活動マップの情報を取得するため，知識活動マップを利用してユーザの活動をより円滑にする機能を提供で

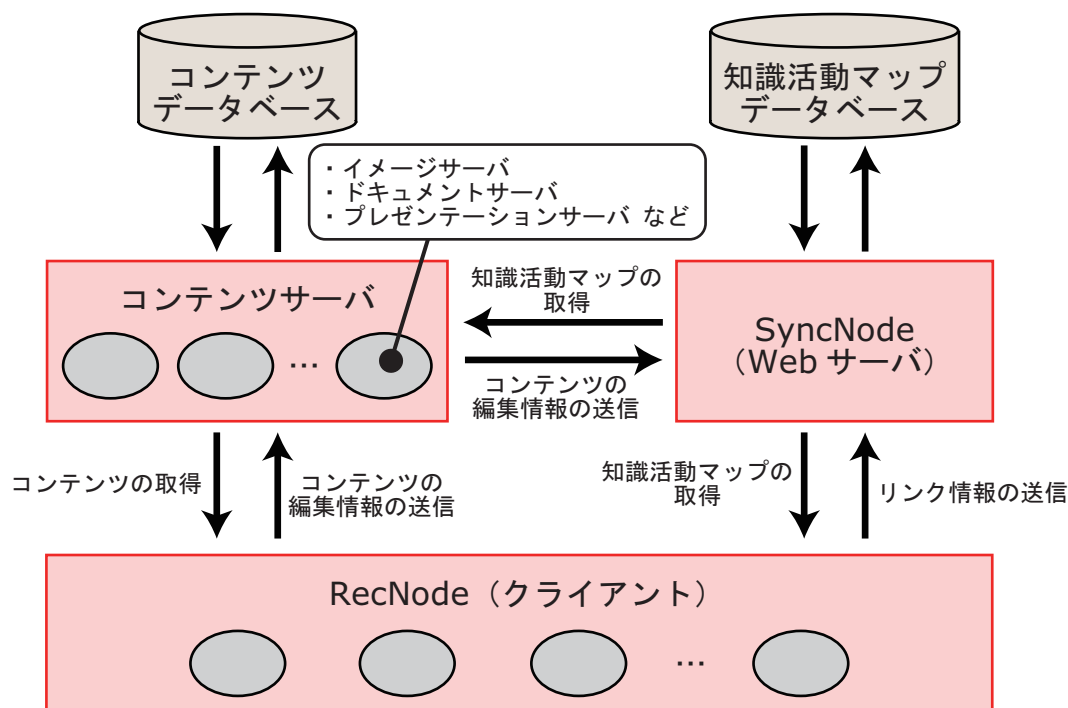


図 2.4: 知識活動支援システムの構成図

きる。RecNode 自体は、ユーザに提供する機能やユーザが使用する状況・環境に合わせて適宜カスタマイズする必要がある。

2.2.5 コンテンツサーバ

コンテンツサーバの特徴としては、次の2点が挙げられる。

- コンテンツやエレメントへの参照形式とアクセス方法
- コンテンツの作成・編集の管理と SyncNode への履歴情報の送信

コンテンツやエレメントへの参照形式とアクセス方法

2.2.3 項では、任意のコンテンツやエレメントを参照できるような柔軟な表現形式を持つポインタが必要であることに言及した。ここでは、プレゼンテーション資料を扱うコンテンツサーバ（以下、プレゼンテーションサーバ）を例として挙げながら、その実装について述べる。

知識活動支援システムでは、Web 上に存在するコンテンツを扱うため、既存の Web 技術との親和性が高い仕組みでコンテンツを識別できることが望ましい。一般に Web サイトの場合は、URI (Uniform Resource Identifier) という形式で Web

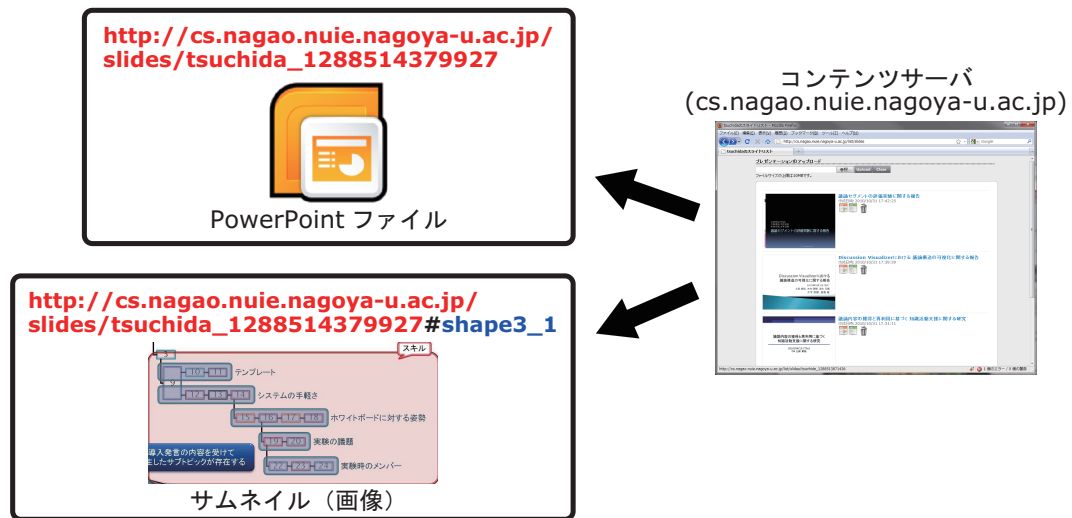


図 2.5: URI を用いたコンテンツへのアクセス

サイトを一意に特定できる．また，最近はブログの普及に伴い，Permalink と呼ばれる，記事の内容が変更されても変化しない固定的なリンクが一般化しつつある．これにより，リンク切れ防止のため URI の内容は一度公開したら原則変更しない，1つの URI には1つのコンテンツのみが対応するという考え方が定着してきた．そのため，本論文でも知識活動マップ内の全てのコンテンツに対して固有の URI を与えている．たとえば，あるプレゼンテーション資料には以下のような URI が与えられている．

```
http://cs.nagao.nuie.nagoya-u.ac.jp
        /presentation/tsuchida_1255414271656
```

この URI のホストに相当する cs.nagao.nuie.nagoya-u.ac.jp はプレゼンテーションサーバそのものを表している．上記の URI に対して，HTTP リクエストで用いられるメソッドの一つである GET メソッドでアクセスすることによって，この URI に該当するプレゼンテーション資料をダウンロードできる．また，コンテンツサーバはコンテンツそのものだけでなく，コンテンツの持つ構造情報も XML（以下，コンテンツ XML）として保持しており，上記の URI に対して，POST メソッドでアクセスすることでその XML を取得できる（図 2.5）．

次に，エレメントへのポインタもコンテンツへのポインタと同様に URI 形式で記述するために，梶らの提案する ElementPointer[19, 20] と呼ばれる表現形式を参考にした．ElementPointer は，コンテンツの URI の後にコンテンツのどの部分を指し示しているかを URI 形式で表現する．以下に ElementPointer の具体例を示す．

```
http://domain1/picture.jpg#epointer(
```

```
http://domain2/dim.rdfs#dim(10px,20px,30px,40px))
```

ElementPointer では、#epointer より前がコンテンツの URI を、#epointer より後ろのスキーマ URI によってそのコンテンツ内のどの部分を指し示しているかを表現している。スキーマ URI に続く引数の順番と意味は RDFS(RDF Schema) により自由に定義できる。この例ではイメージ上における、(10px,20px) を始点として幅 30px、高さ 40px の矩形範囲を表している。

コンテンツサーバでは、ElementPointer を簡略化した形式でエレメントのポイントを記述できるようにした。以下は、前述のプレゼンテーション資料内のとあるスライドページに含まれる一つの図形オブジェクトを参照するための URI の例である。

```
http://cs.nagao.nuie.nagoya-u.ac.jp  
/presentation/tsuchida_1255414271656#shape3_1
```

前述したコンテンツ XML には、各部分要素を表すタグが記述されており、それぞれのタグを一意に区別するための ID が割り振られている（スライドページであれば slide3、図形であれば shape3.1 など）。上記の URI では、#より前がプレゼンテーション資料の URI を表しており、#より後ろが図形情報の ID を表している。

コンテンツの場合と同様に、この URI に対して GET メソッドでアクセスすると、エレメントの内容をダウンロードできるが、その形式は部分要素のタイプによって異なる（図形オブジェクトであればサムネイル画像、テキストオブジェクトであればテキストなど）。また、上記の URI 内の ID と、エレメントに対応するコンテンツ XML 内のタグの属性として付与されている ID は対応しているため、XML ドキュメントの内部ノードを指し示す手段として標準化されている XPath を用いることで、より詳細な内部情報を取得することも可能である（図 2.5）。

プレゼンテーションサーバの場合、コンテンツやエレメントの URI に対して GET メソッドでアクセスすることで、直接他のコンテンツに引用可能な画像やテキストをダウンロードできる。しかし、コンテンツの種類によってはデータをダウンロードするのではなく、コンテンツやエレメントの内容を閲覧するための Web ページが表示される場合もある。たとえば、3 章で述べる会議コンテンツは、テキストや画像、映像が混在する複雑なコンテンツであり、その部分要素をデータとしてダウンロードすることは困難である。その場合は、URI に対応するコンテンツ XML の内部ノードを参照し、引用できる情報を抽出する仕組みが必要となる。コンテンツの作成・編集の管理と SyncNode への履歴情報の送信

コンテンツサーバは、自身が対象としているコンテンツの作成・編集に関する履歴情報を SyncNode へ送信する機能を有する。そのため、コンテンツサーバは、コンテンツの作成・編集に関わる操作を管理する必要がある。その方法には大きく分けて 2 つある。

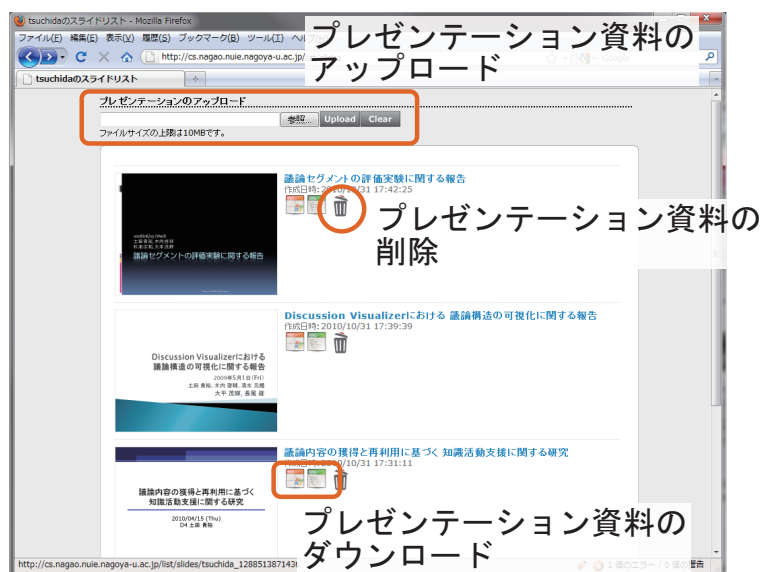


図 2.6: プレゼンテーションサーバ

一つは、コンテンツサーバ内でコンテンツの登録や変更等を行うための機能を提供することである。プレゼンテーションサーバの場合、Web ブラウザでプレゼンテーションサーバへアクセスすると、図 2.6 のようなインターフェースが表示され、プレゼンテーション資料のアップロードやダウンロード・削除を行うことができる。

もう一つの方法は、そのコンテンツに特化した作成・編集用システムを作成することである。特に会議のような実世界インタラクションの内容をコンテンツとして記録する場合は、センサなどを用いてリアルタイムにさまざまな情報を取得し、最後に全ての情報を一つのコンテンツとして統合し、アップロードするような仕組みが必要になると考えられる。もしくは、ユーザが使用する RecNode にコンテンツの作成・編集を行うための機能を持たせることも可能である。

上記のいずれか（もしくは両方）の方法を用いてコンテンツの作成や編集が行われた場合、その履歴情報は SyncNode へ送信され、知識活動マップとして蓄積される。新規に作成されたコンテンツは、知識活動マップの新たなノードとして追加されることになる。

2.2.6 SyncNode

SyncNode は、コンテンツサーバや RecNode から送信されるさまざまな情報を収集し、知識活動マップを蓄積・構築していくための Web サーバである。上記の仕様を実現したすべてのコンテンツサーバからコンテンツに関する情報を収集する

仕組みを持っているが、種類によって構造が異なるコンテンツをすべて記録できるデータベースを設計することは困難である。そこで SyncNode では、すべてのコンテンツにおいて共通のメタデータだけを記録する方法を採用している。具体的なメタデータの項目は下記の通りである。

- コンテンツを識別するための URI
- コンテンツタイプ
- 作成者
- タイトル
- 作成日時
- 最終更新日時

また、引用によって生成されるコンテンツやエレメント間のリンク情報は、以下の項目によって構成される。このリンク情報と先のコンテンツのメタデータを蓄積することで、知識活動マップを構築していく。

- 参照元のコンテンツもしくはエレメントの URI
- 参照先のコンテンツもしくはエレメントの URI
- 作成日時

さまざまなプログラミング言語で実装されたコンテンツサーバや RecNode と、知識活動マップに関する情報を送受信するための標準的な通信を行うため、SyncNode は JSON-RPC²と呼ばれる、軽量な RPC(Remote Procedure Call) のプロトコルを使用している。SyncNode が、コンテンツサーバや RecNode に対して提供しているサービスには以下のものがある。

- コンテンツの編集情報（追加・更新など）の登録
- コンテンツ・エレメントの引用情報の登録
- 知識活動マップに関する情報の取得

コンテンツの編集情報やコンテンツ・エレメントの引用情報は、前述した JSON-RPC を用いて知識活動マップデータベースを更新するためのサービスである。それに対して、知識活動マップに関する情報の取得は、コンテンツサーバや RecNode から集約されたデータを利用して、それら単体では実現できないような新しい応

²JSON-RPC . <http://json-rpc.org/> (2010 年 12 月 11 日)

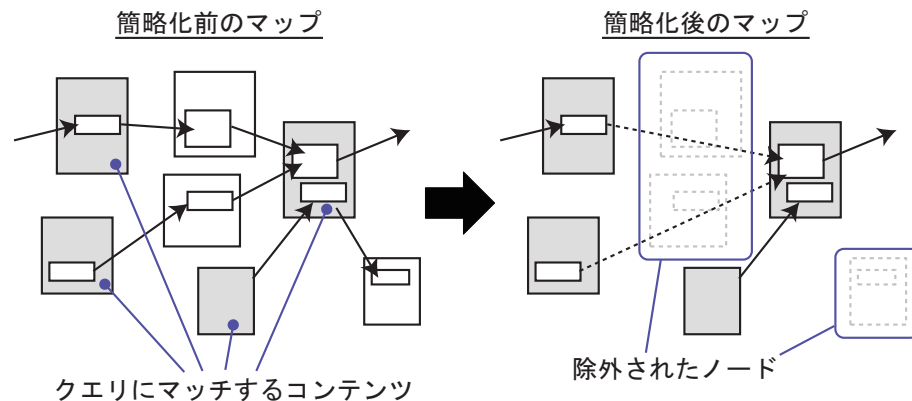


図 2.7: 知識活動マップの簡略化

用を実現するためのサービスである。その機能は、クライアントから送信されたリクエスト情報に応じて、コンテンツのメタデータとリンク情報を合わせたデータを JSON-RPC で返信するというものである。もちろん、知識活動マップデータベースに含まれるデータをすべて送信することも可能だが、クライアントで指定されたクエリに合わせて簡略化した知識活動マップを送信できる。

クライアントが特定のコンテンツに絞った知識活動マップを要求した場合に行われる簡略化の概念図を図 2.7 に示す。左側は簡略化する前の知識活動マップの一部であり、クライアントが指定したタイプに対応するコンテンツには編みかけがされている。このとき、SyncNode は以下の手順でマップの簡略化を行う。

- 1: repeat
- 2: for all *node* in 知識活動マップに含まれる全てのノード do
- 3: if *node* がクエリにマッチしない (除外する) ノードである then
- 4: if *node* を引用しているノードと *node* に引用されているノードのどちらも存在する then
- 5: *node* を引用しているノードと *node* に引用されているノードとの間のリンク情報を作る
- 6: *node* をマップから除外する
- 7: else
- 8: *node* をマップから除外する
- 9: end if
- 10: end if
- 11: end for
- 12: until 除外するノードが存在しない

クライアントは知識活動マップに関する情報を取得するサービスを利用することで、これまで自然言語処理や画像認識のような機械処理を用いて推測していた

コンテンツ間の関係情報を，特別な処理を必要とせずに獲得できる．

しかし，このサービスによって必要以上の情報が獲得できてしまう可能性がある．たとえば，それが個人的なメモ書きであったとしても，SyncNodeに登録されている以上，サービスを通じて取得できる情報の中に含まれてしまう．SyncNodeでは，コンテンツそのものの内容ではなく，コンテンツのメタデータを蓄積しているため，その内容を確認することはできないが，コンテンツのURIを用いてコンテンツサーバを直接参照することで，不正にアクセスされてしまう可能性がある．このようなコンテンツに関する情報の集約・蓄積に伴って発生するセキュリティやプライバシーに関する問題は，今後知識活動支援システムを拡充していく中で対応していく必要がある³．

2.2.7 RecNode

RecNodeの主な機能として以下のものがある．

- ユーザに対するコンテンツやエレメントの引用機能の提供，およびその操作履歴からのリンク情報の獲得
- 前述のSyncNodeが提供するサービスを用いたリンク情報の送信

後者の機能が，ユーザの操作に関係なく暗黙的に実行されるものであるのに対して，前者の機能は，ユーザの操作と直接関係する機能である．引用機能そのもののユーザビリティが低いと，RecNodeを使用するモチベーションが低下しデータが収集できない可能性があるため，できるだけユーザが日常的に行っている操作に近い自然な操作で引用ができることが望ましい．その方法には，コピーアンドペースト機能を利用するものとそれ以外のものがある．

コピーアンドペーストは，われわれが日常的に使用している編集機能であり，引用元の箇所を選択しコピーすることで，クリップボードにデータが転送され，引用先の箇所でペーストすることでそのデータが挿入される機能である．クリップボードにはさまざまな形式のデータを転送することができ，ペーストする際はクリップボードに含まれる形式の中から，引用先のアプリケーションにもっとも適したものを選択できる．たとえば，表計算ソフト Microsoft Excel でテーブルデータをコピーした場合，クリップボードにはオブジェクト形式やテキスト形式で表現されたデータが転送される．それをテキストエディタにペーストした場合はテキスト形式，プレゼンテーションソフト Microsoft PowerPoint 上でペーストした場合はオブジェクト形式が選択され，それぞれの形式に対応したデータが挿入される．この仕組みを利用してリンク情報を取得する方法は次の通りである．

³解決策の一つとして，コンテンツサーバ側でコンテンツの公開範囲を設定できるようにすることが挙げられる．

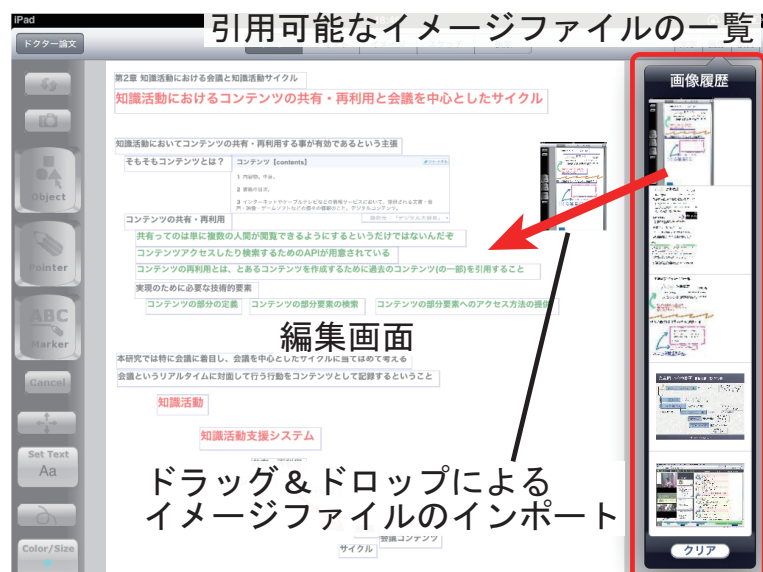


図 2.8: RecNode におけるコンテンツ・エレメントのインポート

- RecNode で表示しているエディタ内で、コンテンツのエレメントを選択・コピーしたときに、コピーされたエレメントの URI と内容を含むデータをクリップボードにコピーする。
- 別の RecNode（もしくは同じ RecNode 内の別のウィンドウ）上にペーストする際に引用元のエレメントの URI とデータを取得し、データは引用先のエレメントとして挿入し、URI は挿入されたエレメントの URI とペアにしてリンク情報を生成する。

コピーアンドペーストを利用した方法は、引用元および引用先のコンテンツを開いているエディタがどちらも RecNode 上で実装されている必要がある。それに対して、引用先のエディタだけが RecNode 上で実装されている場合は、引用できるコンテンツ・エレメントを検索し、自身のエディタにインポートする必要がある。図 2.8 に例を示す。このインタフェースは、テキストやイメージ、図形などを自由にレイアウトし、アイデアの整理を支援するためのインタフェースである。そして、インタフェースの右側に引用可能なイメージファイルの一覧が表示されており、その中から必要なイメージファイルをドラッグ&ドロップで編集画面にインポートすることができ、インポートされたイメージファイルを自由に移動・リサイズを行うことができる。

2.3 知識活動における会議と知識活動サイクル

2.3.1 知識活動における議論内容の再利用とその意義

2.1 節で、知識活動が形式知と暗黙知の相互作用によるスパイラルとして表されることを述べたが、そのスパイラルの中には調査や実験・検証、会議といったさまざまなタスクが内在する。たとえばテーマを実現するために必要な情報を Web や書籍、論文といった文献を調査したり、アイディアの妥当性を確認したりするために実験を行う。そこで得られた実験の結果を検証することによって新たな問題点が見つかったり、新たなアイディアや知識が生まれたりする。こうして蓄積されたアイディアや知識を論文やスライドなどのドキュメントにまとめ、他者に向けて発表し、議論することで活動に対するフィードバックを獲得できる。そして、フィードバックを活かして新たな調査や実験・検証などを行っていく。このように知識活動に内在するタスクは、独立に行われるのではなく、相互に関係しながら繰り返し行われていく。

本研究では、知識活動に内在するタスクの一つである会議に着目する。ここで述べる会議とは、自身がこれまでに行ってきた活動の内容を他者に向けて発表し、さらに自分だけでは解決できないような問題点等について意見を出し合う行為をさす。例としては大学研究室におけるゼミが挙げられる。会議は以下の2点の理由から知識活動の中で重要な役割を果たしていると考えられる。一つは自身がこれまでの知識活動を通じて培ってきたアイディアや知識を他者に向けて公表する点であり、もう一つは議論に参加している人間がさまざまな視点から意見の交換を行う点である。

知識活動を通じて創造されたアイディアや知識は、活動を行っている本人だけが所有するのではなく、社会に向けて提供することによって、さらなる価値を持つことになるだろう。そのため、複数の人間を集め、アイディアや知識を公表する場を提供するという点で議論を行うことは非常に意義のあることである。また、発表を行うための話し方や議論展開の方法、発表に用いるスライドの作成方法など、よりよい議論を行うためのノウハウを集めた書籍が数多く出版されている。そして、Microsoft PowerPoint や OpenOffice.org Impress に代表されるような、発表に使用するスライドの作成や発表そのものを支援するプレゼンテーションソフトも広く普及している。このようによりよい発表を行おうという試みも、知識活動において議論が非常に重要なものであるということを裏付けているだろう。

ゼミや学会発表では、議論に参加している人間は類似するテーマに関心を持っていることが多く、それぞれ独自の考え方や方法に基づいて知識活動を行っているだろう。つまり、発表のテーマに関してさまざまなアイディアや知識を持った人間が、議論の中でさまざまな質問やアドバイスを行うことによって、発表者はその後の知識活動をよりよくするためのフィードバックを獲得できる。たとえば、

「あなたのやっている研究に関係のあるものとして、こういう研究があるので参考にしたい」と、自身の知識活動に関連のあることを指摘されれば、その内容を調査することによって、新しい知識の獲得や自身の知識活動の社会的な位置づけの把握をできる。このように議論で行われた内容は、その後のタスクにさまざまな影響を及ぼす。

しかし、会議に関する経験的な問題点として、過去に行った議論と同じ内容の議論を繰り返してしまったり、過去に指摘された点がその後の活動に反映されていなかったりすることが挙げられる。その原因として考えられるのは、時間の経過と共に過去に行った議論の内容を忘失してしまうことである。その時は先の話として保留しておいた議論が次第に忘れ去られてしまい、結果として真にその議論が必要な時に同じ内容について冗長な議論を行ってしまう可能性がある。また、当時はあまり本質ではないと思っていた議論が、後から視点を変えてみると思いがけない価値を持ったものであると気付くこともあるかもしれない。このように再利用すべき価値を持つ議論内容を何らかの手段で記録することで再利用できるようになる。

2.3.2 DRIP サイクル

DRIP サイクルは、会議を中心に以下の4つのフェーズから構成される。

- 議論 (Discussion) フェーズ
- 再認 (Reflection) フェーズ
- 探究 (Investigation) フェーズ
- 集約 (Preparation) フェーズ

議論 (Discussion) フェーズ 前述した議論そのものに相当するフェーズであり、これまでの知識活動から生まれたアイデアや知識を他者に向けて発表し、自分だけでは解決できなかった、もしくは他者の意見を参考にしたい点などについて、参加者が多角的な視点から質問や意見などの発言を行うことによって、発表者はフィードバックを獲得する。その際、議論内容を記録した会議コンテンツを作成することで、議論から得られたフィードバックをその後の活動で利用できるようになる。

再認 (Reflection) フェーズ 作成された会議コンテンツには、「自分にとって有益な議論は何なのか、そして、その議論がなぜ有益なのか」という人間の解釈が含まれていない。この情報は、目的とする議論を探し当てる際に重要な情報であ

る。一般的なコンテンツの検索手段としてキーワード検索があるが、時間が経過するほど、議論内容を忘失し適切なキーワードを入力することが困難になる。そのため、このフェーズでは、会議コンテンツの中から重要な議論を引用しながら、自分の解釈を加えながら議論内容の整理を行う作業（本研究ではリフレクションと呼ぶ）を行う。リフレクションを行うことによって、会議コンテンツの再利用を行いやすくなる。また、このフェーズには議論内容の整理を通じて、今後の活動に関するタスクを明確にする作業も含まれる。

探求（Investigation）フェーズ 議論中に指摘された情報不足を補うために調査を行ったり、重要なアドバイスに基づいて新たなアイデアを生み出したりするように、状況に応じて会議コンテンツを適切に利用しながらタスクを行う。タスクを通じて生み出されたアイデア・知識と、そのとき利用されていた議論を関連づけることによって、議論をどれだけ活用しているかを知ることができる。一つの議論に対するアイデアや知識の量や質によって自身の活動の成熟度を測ることができ、見落としている議論の存在を知ることによって、より広い視野を持つことができる。

集約（Preparation）フェーズ 探究フェーズで生み出されたアイデアや知識を、他者に向けて発表し、議論を行うことによって知識活動の内容が組織内で共有され、新たなフィードバックを得ることができる。このようなフィードバックを得るためには自身の知識活動の内容を他者に正確に伝達する必要がある。そのためこのフェーズでは、知識活動を行っている人間が自身の活動内容を把握し、発表資料としてまとめることが求められる。そして、作成された発表資料を用いることによって新たな議論が生まれ、さらなるフィードバックを獲得できる。

これら4つのフェーズが、図2.9のように繰り返し行われることによって、多くのアイデアが創造され、知識として具体化・理論化される。DRIPサイクルに関連する概念にPDCAサイクル[24]がある。PDCAサイクルは、企業の生産管理や品質管理を改善するための手法であり、実績や予測に基づいて計画を立てる段階であるPlan（計画）、計画に沿って実際に業務を行う段階であるDo（実施）、業務内容が計画に沿っているかどうかを確認する段階であるCheck（評価）、そして計画に沿っていない箇所を改善する段階であるAct（是正処置）の4段階から構成される。この4つの段階がDRIPサイクルの各フェーズと厳密に対応するわけではないが、DRIPサイクルは、会議を中心とする知識活動に合わせてPDCAサイクルを最適化したものと捉えることができる。なぜなら、知識活動に関する計画の作成（再認フェーズ）、計画に基づく様々なタスクの遂行（探求フェーズ）、議論を通じての活動内容に関する評価（議論フェーズ）という流れは、PDCAサイクルと類似している点が多いからである。

DRIPサイクルにおいて最も特徴的な点は再認フェーズである。これまでに、

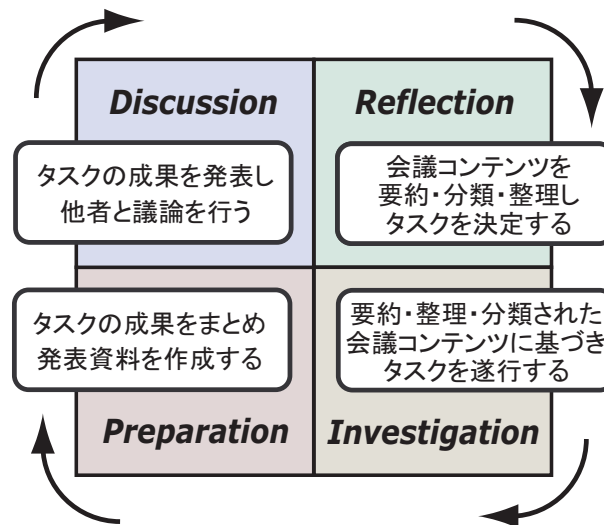


図 2.9: DRIP サイクル

発表に関するノウハウとして、発表中にメモを取るという行為が暗黙的に行われてきた。相手の発言を自分の頭の中で整理し、自分にとって重要な個所を書き残すことによって、それ以降の活動の中で再利用することが可能になる。そして、記録されたメモは「ディスカッションマイニング関連」や「個人研究関連」のように活動の内容に合わせて分類されることによって、必要に応じて探し出すことができる。

しかし、議論を行っている最中にメモを書くという作業は負担が大きいため、必要な議論内容をすべて記録することは困難である。そのため、書き残されたメモは、単語の羅列であったり不可解な文章であったりと必ずしも事後の利用に適した形式で残されとは限らない。結果として見返したときにその内容が把握できない可能性がある。さらにメモの分類方法も、内容や記録された日付順などさまざまであり、状況に応じて適切に変更できることが望ましいが、紙でメモを行っている場合は蓄積されたメモの分類方法を変更することは容易ではない。

一方、議論内容を議事録として記録することが広く行われている。一般に、人間が発言を聞き取りながらその内容を記録する行為は負担が大きいため、情報技術を駆使して議事録の作成を行う研究が数多く存在する。このような手法は議論内容を網羅的に記録できるという利点がある。しかし、もともと議事録は共有するという観点に基づいて作成されているため、そこにはメモのように発表を行った人間の解釈が含まれていない。そのため、過去に重要だと思った議論を再度閲覧したいときは、まず、その議論が含まれている議事録を検索する。さらに目的とする議論が議事録のどの位置にあるかを見つける必要がある。このように、議論に対して解釈が付与されていないと、同じ議論を閲覧する度に、冗長な作業を

繰り返すことになる。

再認フェーズでは、議論フェーズで発表・議論が行われた後に、発表者が議事録に解釈を付与することによって、これまでは誰が閲覧しても同じ内容を提示するコンテンツであった議事録を、プライベートなものとして編集する。議論内容を網羅的に記録した議事録を、発表中に記録するメモのように個人化することで、必要とする議論を探し出すという作業を効率的に行うことが期待できる。

再認フェーズに関する考察 一般に、時間の経過と共に議論の内容を忘れてしまうため、再認フェーズは議論フェーズからできるだけ時間を空けずに行うことが必要であると思われる。そこで、著者と同じ研究室に所属する学生を対象に予備調査を行った。具体的には、研究活動に関するゼミ発表・議論が終わった直後と1週間後に、下記の項目に関するアンケートを行った。

- 議論の中で重要だと思われる発言はどれか
- その発言が重要であると判断した根拠は何か

その結果、発表直後に行った時に比べ、発表から1週間後に行った時のほうが重要であると判断した発言の数が少ないことが確認された。また、それぞれの作業が終わったときに感想を記述してもらった。その中には、「(1週間後に行ったときは内容を)ほとんど覚えていない」のように議論内容を忘れかけていると推測されるコメントがあった。このことから、再認フェーズは議論フェーズの直後に行うことが望ましいと思われる。

また、この予備調査では、発言単位に対して重要性の判定やその根拠を記入してもらったが、調査結果の中には、その作業を行う対象は発言単位ではなく、意味的にまとまった話題単位のほうがよいというコメントがあった。これは、一つの発言だけではなく、その発言の派生元となる発言や、その発言から派生した発言まで含めて議論内容を見ることによって、より深い内容理解につながるからであると推測される。

この他にも、1週間後に作業を行った際、発表直後に行った時とは異なる視点で同じ議論を捉えることがあるというコメントがあった。これは、発表直後から1週間の間に同じテーマに関する別の発表が行われており、そのとき行われた議論の内容が1週間後の作業に影響を及ぼしたためだと考えられる。このような結果を踏まえると、知識活動の中で繰り返し議論を行うことによって、過去の議論に対する考え方は常に変化し続けるのではないかという仮説が考えられる。そのため、再認フェーズでは重要な議論に対して、アクセスしやすくするための情報の付与だけを行い、変化し続ける議論に対する考え方は探究フェーズで付与・変更することを考えた。

以上の考察を踏まえ、再認フェーズでは、話題単位でセグメントされた議事録に対して、重要なものであればアクセスしやすくするための情報を付与するとい

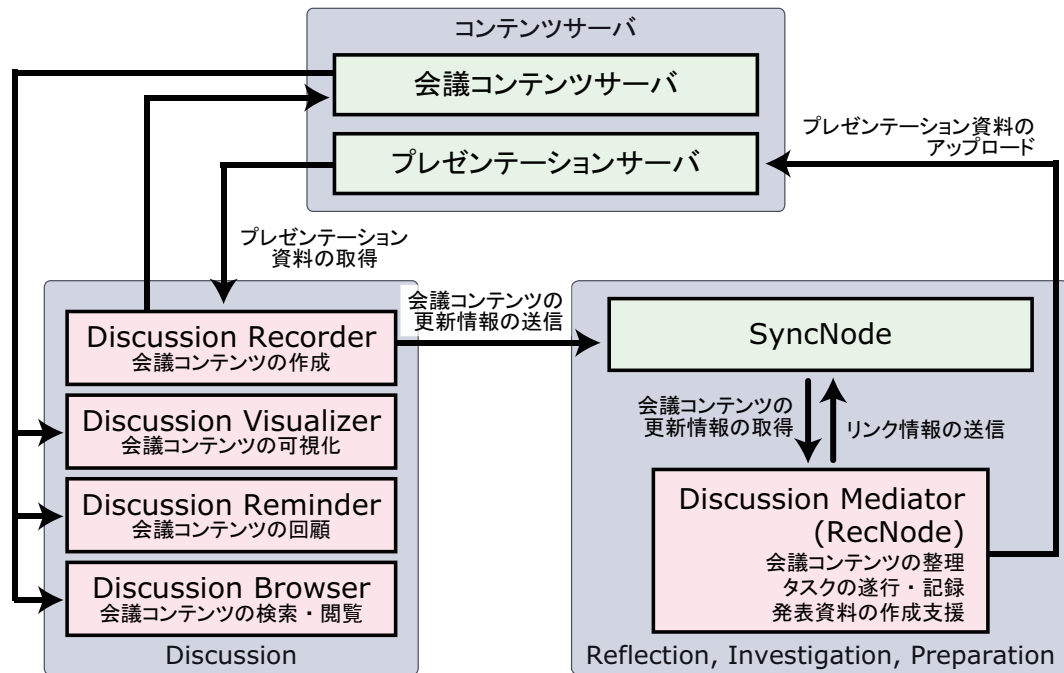


図 2.10: DRIP システムの構成

う作業を行うものとした。アクセスしやすくするための情報の例としては、議論内容を端的に表すようなキーワードや「調査すべきことを探す」「実装方法を考える」といった議論内容の利用目的が挙げられる。

2.3.3 ディスカッションマイニングによる DRIP システムの構築

前節で述べた DRIP サイクルの各フェーズを統括的に支援するシステム（以下、DRIP システムと呼ぶ）を知識活動支援システム上に構築するため、本研究では、ディスカッションマイニングと呼ばれる技術を用いた。ディスカッションマイニングは、人間同士の知識交換の場である会議から実世界情報を獲得・構造化し、再利用可能な知識を抽出する技術である。DRIP システムが主に扱う点は下記のとおりである。

- 会議情報の獲得による会議コンテンツの作成
- 会議コンテンツを用いた議論の円滑化支援
- 会議間の活動における会議コンテンツの再利用

図 2.10 に DRIP システムの構成を示す。DRIP システムは、DRIP サイクルの各フェーズに合わせたアプリケーションから構成される。議論フェーズで利用する

アプリケーションには、会議コンテンツの作成を行う Discussion Recorder、その場で会議コンテンツを可視化する Discussion Visualizer や過去の会議コンテンツの検索・回顧を行う Discussion Reminder、作成された会議コンテンツの検索・閲覧を行うための Discussion Browser がある。

複数の人間が一堂に会して会議を行うため、会議中に用いる Discussion Recorder, Discussion Visualizer, Discussion Reminder は共有スペースで使用されることを想定している。それに対して、議論フェーズ後に行われる再認フェーズや探求フェーズ、集約フェーズでは、システム利用者が個人用のスペースで作業するため、Discussion Mediator と呼ばれるクライアントアプリケーションを提供している。Discussion Mediator は、知識活動支援システム内の RecNode に相当するアプリケーションであり、以下に示した支援を行う。

- (再認フェーズ) 会議コンテンツの整理
- (探求フェーズ) 整理した議論内容に基づくタスクの遂行およびアイデアや知識の記録
- (集約フェーズ) 蓄積されたコンテンツの再利用に基づくプレゼンテーション資料の作成

DRIP システムでは、上記のアプリケーションの他に 2.2.5 項で述べたプレゼンテーションサーバも利用する。Discussion Mediator で作成されたプレゼンテーション資料はプレゼンテーションサーバへアップロードされる。そして、Discussion Recorder を用いた会議を行う際にダウンロードされ発表に利用できる。

2.4 本章のまとめ

本章では、知識活動を議論中心の複数フェーズからなるサイクルとして捉え、DRIP サイクルと呼ばれるサイクルを提案した。DRIP サイクルは、議論を行うことで参加者からさまざまな意見やアドバイスを獲得する議論フェーズ、獲得した議論内容を後から効率的に再利用するために整理する再認フェーズ、整理された議論内容に基づいて新たなアイデアや知識を創造する探究フェーズ、蓄積されたアイデアや知識を再利用して次の発表で用いる発表資料の作成を行う集約フェーズから構成される。そして、このサイクルを統括的にサポートする DRIP システムを提案した。この DRIP システムは、DRIP サイクル内の各フェーズに合わせたアプリケーション群から構成されている。

次章では、議論フェーズにおいて会議コンテンツを作成するための仕組みである Discussion Recorder と、Discussion Recorder によって作成された会議コンテンツの分析について述べる。

第3章 会議コンテンツの作成

さまざまなタスクにおける議事録の効率的な再利用を実現するためには、議事録の記録方法を工夫する必要がある。これまでも議事録を作成するシステムに関する研究は数多く行われている [2, 33, 38]。議論の詳細情報の取得方法もテキストだけでなく、発表に用いられた資料や、議論の様子を記録した映像・音声を組み合わせるなど多岐に渡る。しかし、これらのシステムは議事録を作成することに重点を置いており、議事録を用いた応用は検索や要約など、実際にどのように使用されるかまで考慮されていない。

会議記録から話題を抽出する古田ら [39] や栗原ら [26] の研究に見られるように、一般に議事録は話題単位で閲覧することが多い。しかし機械による意味解釈は非常に困難であるため、テキストや音声を用いた話題抽出の精度は決して高いとは言えない。そこで我々は、議論内容をテキストやビデオを組み合わせたマルチメディア議事録として半自動的に記録する、Discussion Recorder と呼ばれる技術を研究・開発している [13]。

Discussion Recorder では意思決定を目的とするのではなく、発表を主体とした議論を行う会議を対象としている。対象とする会議ではモデレータとなる発表者、その発表を聴き意見を述べる参加者、そして会議の記録を行う書記がいる。会議のモデレータは、発表資料としてスライドをプロジェクタで投影して発表を行うことを前提にしている。実際に Discussion Recorder を利用しているゼミの風景を図 3.1 に示す。

3.1 システム構成

会議を行うミーティングルームは図 3.2 のような空間を想定している。ミーティングルームには音声を記録するためのマイクが設置されている。会議の詳細な文脈を記録するために、1 台のパンチルトカメラが設置されている。また、スライドを投影するスクリーンを記録するための固定カメラが 1 台、そして発表者の様子を記録するための固定カメラが 1 台設置されている。また、Discussion Recorder ではミーティングルームの他に、議事録の作成・管理を行うための議事録サーバが用意されている。

会議中の発言に関する情報を記録するための設備として、参加者全員がポイン

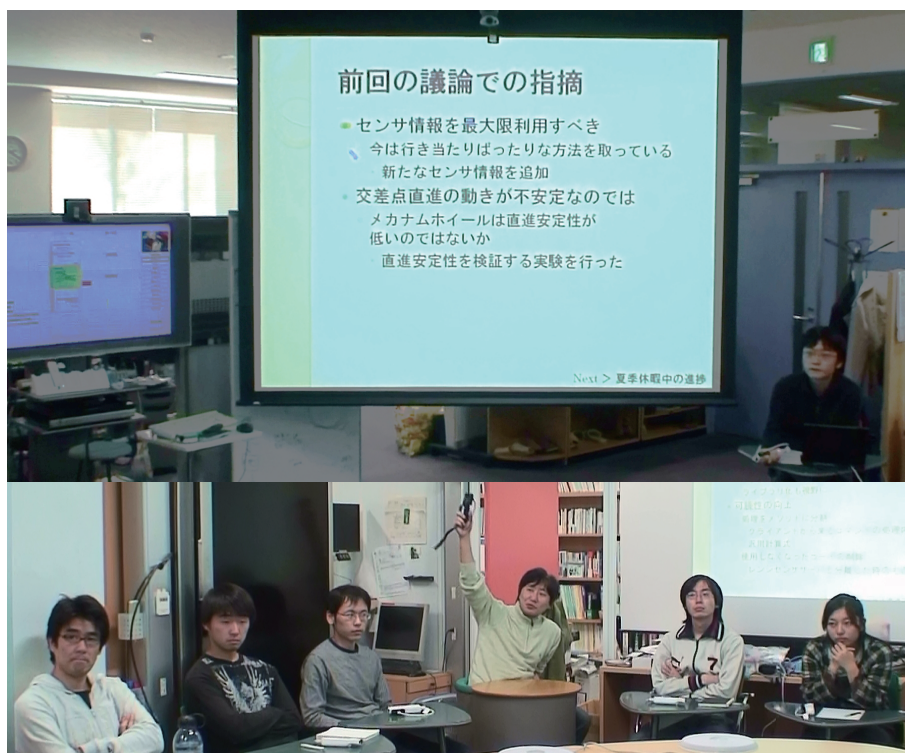


図 3.1: Discussion Recorder を利用しているゼミの風景

タリモコンと呼ばれるデバイスを所持し、各参加者の頭上には赤外線信号を発信する IR バーが設置されている (IR アレイ)．会議において発言する際に、発言者はポインタリモコンと頭上の IR バーを一対一対応させることで、発言時の情報を送信する仕組みになっている．発言に関する情報は随時システムに送信され、書記の発言内容入力の手助けを行う．

パンチルトカメラは、会議中ディスカッションルーム内をスウィングしながら参加者の様子を記録している．また、発表者以外の任意の参加者が発言を始めたとき、ポインタリモコンと連動し、パンチルトカメラがその方向に固定され、発言者の様子を撮影する．発言終了後には再び会場内を記録するためにスウィングモードに切り替わる．スクリーン用カメラは、スクリーンに投影されるプレゼンテーションに用いたスライド、デモ、その他の参考資料の様子を記録する．

カメラやマイクで会議の詳細な文脈を記録した音声・映像は MPEG-4 形式で映像・音声データベースに保存される．発表者が入力したスライド情報、書記が入力したテキスト、参加者のデバイスを使って獲得したメタデータは、議事録 XML として XML データベースに保存される．

このようにして記録された議事録は XML と MPEG-4 によるマルチメディア議事録としてデータベースに記録される．記録された議事録は Web ブラウザを用い

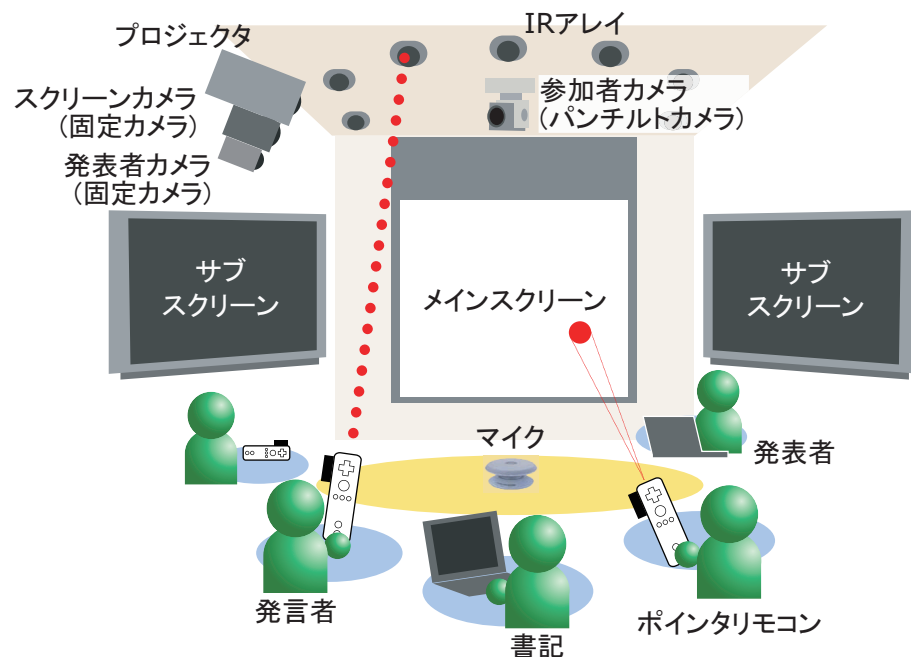


図 3.2: ディスカッションルーム

て容易に閲覧できる。

3.2 発表情報の取得

発表者は、図 3.3 に示されるブラウザベースの専用ツールを使用してプレゼンテーションの操作を行う。まず、会議開始時に専用ツールから発表に用いるスライドファイルを議事録サーバにアップロードする。アップロード時には発表の内容を表すキーワードや、サーベイレポート、研究進捗といった発表のカテゴリを選択する。アップロードが終了すると、会議が開始され開始時刻が議事録サーバに送信される。発表者が専用ツールを用いてスライド操作を行うと、スライドの切り替えタイミングやスライドアニメーションの表示タイミングが随時サーバに送信され、記録される。また、スライド以外の資料(デモや Web の参照)を用いてプレゼンテーションを進める場合には、資料を追加することもできる。会議終了時には、終了時間が議事録サーバに送信・記録される。

3.3 発言情報の取得

Discussion Recorder を利用する参加者(発表者、書記を含む)は、ポインタリモコンというデバイス(図 3.4)を利用する。参加者は発言を行う際に、ポインタリ

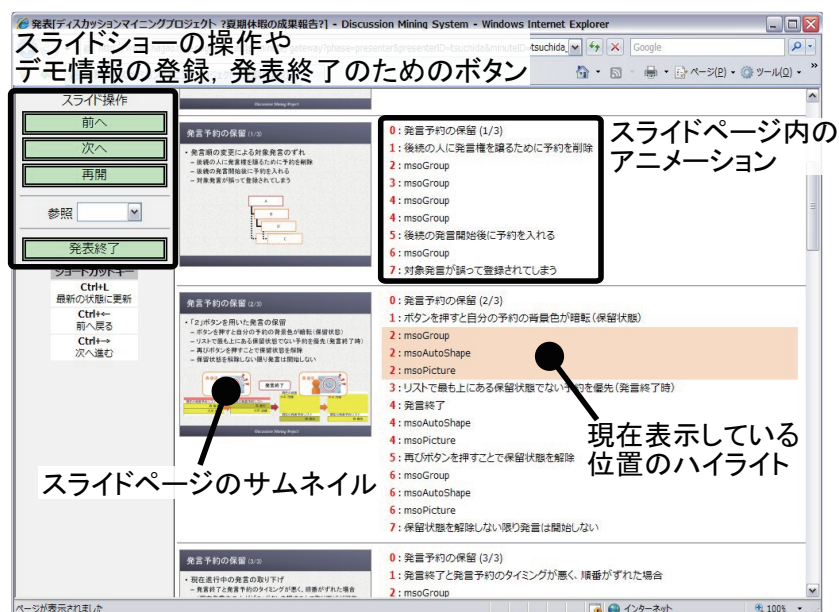


図 3.3: 発表者用インタフェース

モコンを上に向けてから発言を行う。発言者の頭上には赤外線信号を送信するための IR バーが設置されており、IR バーから送信される信号には座席位置を識別するための ID が含まれている。この信号を受信したポインタリモコンは、座席位置だけでなく、ポインタリモコンの利用者（この場合は発言者）の ID や発言の種類（以降、発言タイプと呼ぶ）も含めた情報をサーバに対して送信する。議事録サーバには、これらの情報に加えて受信した時刻が送信・記録される。また、発言の終了時刻はポインタリモコンのボタン操作によって入力できる。

書記は図 3.5 に示される Web ブラウザベースの専用ツールを用いて議論の構造化と発言内容の記録を行う。また、参加者を撮影するカメラの制御や、データベースへの登録作業などもこの書記ツールから行うことができる。書記ツールは前述のポインタリモコンと連動しており、参加者が入力した情報が随時追加されていく。ポインタリモコンから情報を発信すると、書記ツールに発言者と発言タイプが付与されたノードが生成される。書記はこのノードを選択し、テキストを入力することで発言内容を記録できる。

3.4 議論構造の取得

Discussion Recorder で取得する発言情報には、「導入」と「継続」の2つのタイプがある。新しい話題の起点として発言する際には「導入」を、直前までの発言の内容を受けて発言をする際には「継続」を、発言開始時のポインタリモコンの



図 3.4: ポインタリモコン

向きによって入力する。Discussion Recorder は、継続発言と派生元の発言との間にリンク情報を生成し、記録する。これを繰り返し行うことによって、図 3.6 のように導入発言に継続発言が連なる形式の発言集合が複数生成される。本研究ではこの発言集合を議論セグメントと呼ぶ。一つの会議コンテンツ内に複数の議論セグメントを作成することで、会議で行われた議論を話題単位で閲覧できる。

Discussion Recorder では、クロストークがないように参加者の発言の順番を制御するための仕組みとして、発言予約機能を備えている。誰かの発言中にポインタリモコンを上げた場合は発言予約リストに加えられ、直前の発言が終了すると自動的に発言権が移る。複数の発言予約が存在する場合、発言の順番は以下のルールに基づき変動する。

- 発表者の入力した予約は他の予約より優先される
- 導入と継続の予約が両方存在するときは、現在の話題を継続して行えるように継続の予約が優先される

この発言予約機能は、発言の順番を制御するだけでなく、より人間の意図を反映した議論構造を作成するためにも利用される。発言予約機能を利用せずに、発言タイプ（「導入」「継続」）だけを利用した場合、作成される議論構造は導入発言を起点とするリスト構造となる。しかし、一つの発言の内容に対して複数人がさまざまな視点から意見を述べた場合の議論構造は、単純なリスト構造ではなくツリー構造であると考えられる。そのため、誰かが発言を行っている最中に「継続」の予約が追加された場合、その継続発言と現在行われている発言との間にリンク情報を生成する¹。つまり、発言中に複数の「継続」の予約が追加された場合、一

¹発言者がいないときに「継続」の予約が追加された場合は、直前の発言に対してリンク情報を生成する。

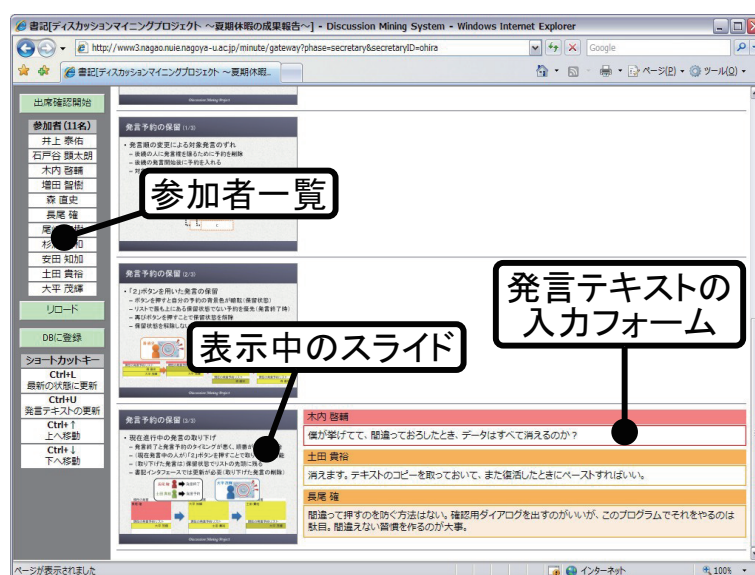


図 3.5: 書記用インタフェース

つの発言に複数の継続発言が連なるツリー構造が自動的に作成される。

3.5 会議に関するメタデータの取得

ポインタリモコンは、発言の開始・終了時間の入力だけではなく、発表者や参加者の発言に対する自分のスタンスを随時入力できる。ポインタリモコンで入力することのできるスタンスには、「賛成」、「反対」、および後述する「マーキング」の3種類がある。参加者はスタンスに応じたボタンを押下することで議事録サーバに自分のスタンスの情報を送信する。

また、ポインタリモコンは、メインスクリーンに向けることによって、レーザーポインタのように画面上の任意の箇所をポイントしたり、線や図形を描画したりできる。これにより発言者がスライドのどの部分に対して言及しているかが分かるようになっている。また、参加者全員がポインタリモコンを所持しているため、図 3.7 のように、複数人が同時に図形を描画することも可能である。

3.6 会議コンテンツの妥当性に関する評価

Discussion Recorder では半自動的に会議コンテンツの作成を行う。しかし、人間が情報の入力を行うため、情報の欠落や誤った情報の入力などのヒューマンエラーが混在する可能性がある。また、議論をできるだけ妨げないような入力方法

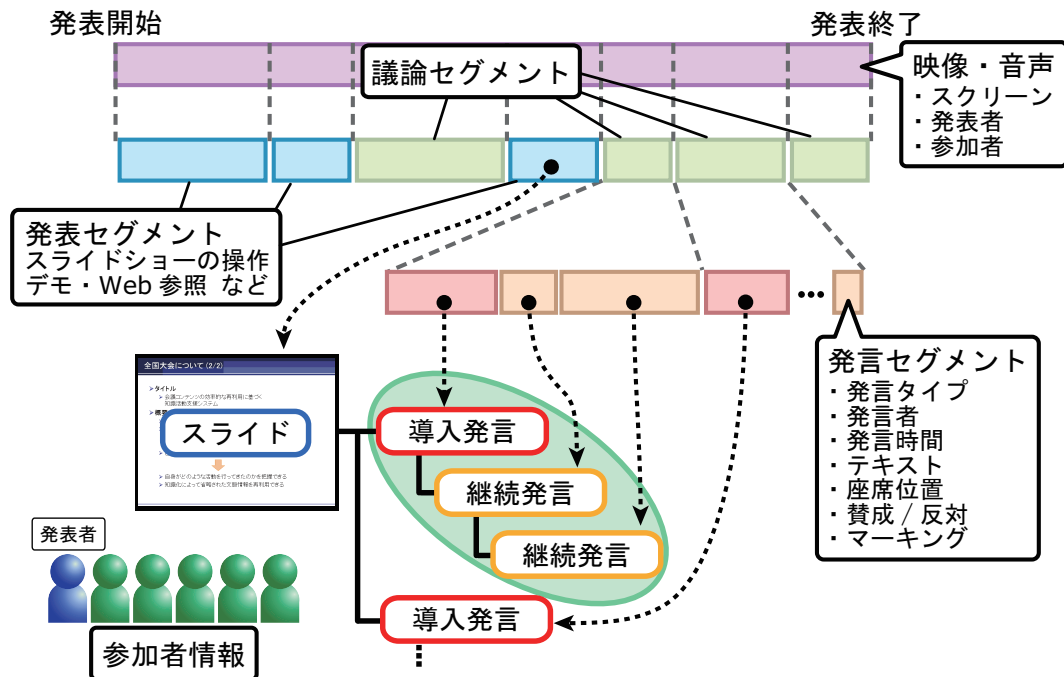


図 3.6: 会議コンテンツ

を採用しているため、取得されたメタデータにユーザの意図が一部しか反映されていない可能性もある。Discussion Recorder で取得しているメタデータの中でも、ポインタリモコンの上げ下げによって取得される議論構造、および書記によって入力される発言テキストは、特に人間による解釈に大きく影響されるものである。そこで本研究では、Discussion Recorder で獲得した議論構造および発言テキストの妥当性の評価を行った。

3.6.1 議論構造の妥当性の評価

Discussion Recorder では、発言開始時に取得される発言タイプ（「導入」・「継続」）や発言予約機能によって議論構造を生成している。しかし、自分がこれから行う発言のタイプが導入なのか継続なのかを迷ったり、予約を入れるタイミングを逸して意図したリンク情報が付与されないなど、生成される議論構造には誤りが混在する、もしくはユーザの意図が一部しか反映されていない可能性があるため、議論構造の妥当性の評価を行った。

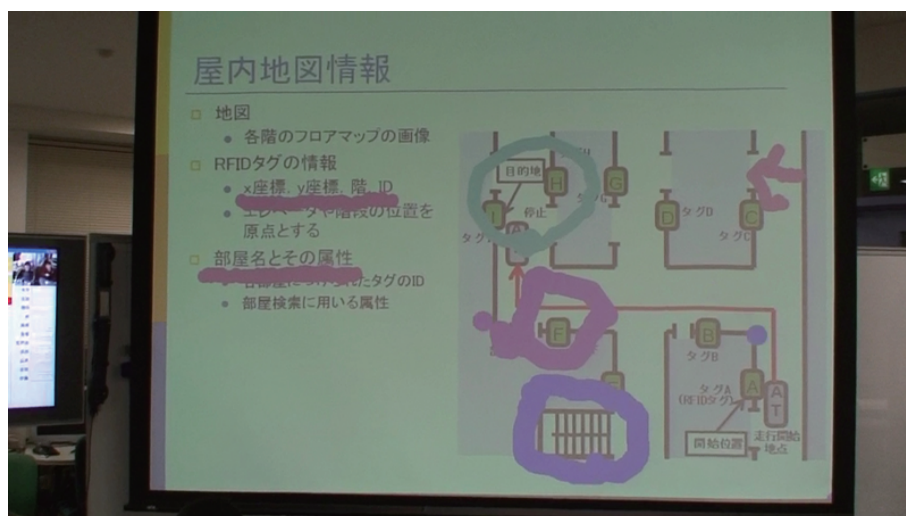


図 3.7: 複数参加者による図形の描画

評価に用いたデータセットと評価方法

2007 年度以降に作成された会議コンテンツにおける議論セグメントの中から、発言数が多い上位 18 個の議論セグメントを対象とした。なお、18 個の議論セグメントに含まれる全発言数は 199 発言（うち継続発言は 181 発言）であり、議論セグメントごとの発言数の平均は 11.1 発言、議論セグメント内の発言者数の平均は 4.6 名であった。そして以下の項目に関して正解データを作成し、比較することで評価を行った。

(1) 新たな話題を提示している継続発言

経験的な問題として、議論が長くなるにつれ話題が次第にずれていくということがしばしば発生する。そのため、発言数の多い議論セグメントに含まれる個々の発言内容を吟味していくと、その議論セグメントの親である導入発言の内容や意図とは関連の弱い継続発言が含まれると考えられる。そのため、議論セグメントの中で新たな話題を提示している継続発言の数を調べることによって、継続発言の妥当性を検証する。

(2) 親発言の異なる継続発言

発言予約機能は、対象となる発言が行われている時点で発言予約を入れることによって発言間のリンク情報を付与する。そのため、対象となる発言が終了した後に発言予約を入れても、すでに別の発言が行われていた場合、正しいリンク情報が付与されない可能性がある。そのため、親発言の異なる継続発言の数によって発言予約機能の妥当性を評価する。

(3) 複数リンクを持つ継続発言

現在の発言予約機能では、親発言となる発言は一つに限定されているが、これ

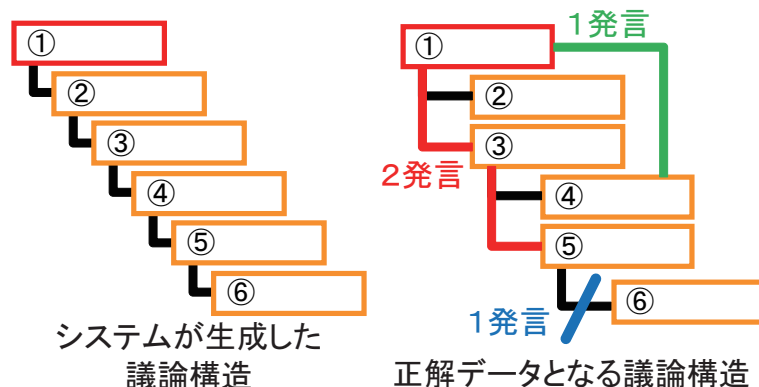


図 3.8: システムが生成した議論構造と正解データとの比較

まで行われた議論をまとめるような発言を行う際には1発言だけでなく、複数の発言を参照することが考えられる。そのため、複数リンクを持つ継続発言の数を調べることで、ユーザの意図が正しく反映されているかどうかを調べる。

図 3.8 はシステムが生成した議論構造と正解データとの比較の例である。この例において (1) に該当する発言は⑥であり (2) に該当する発言は③と⑤の2発言、(3) に該当する発言は①と③との間にリンクを持つ④の1発言となる。

正解データは以下の手順によって作成した。まず、議論セグメントの親となる導入発言の内容や意図がその後の継続発言に正しく反映されているかを確認するため、対象の議論セグメントの導入発言者に自身の発言の内容・意図と関連のない継続発言を選択してもらった。次に、同研究室の大学生・大学院生3名に導入発言者の意図を踏まえた上で、ビデオを視聴しながら(1)から(3)の項目について正解データを作成してもらった。最後に、個々に作成された議論構造を照合しながら合議によって最終的な正解データとした。

評価結果と考察

Discussion Recorder によって生成された議論構造と正解データとの比較を行った結果を表 3.1 に示す。

(1) 新たな話題を提示している継続発言の傾向

表 3.1 から、ほぼ全ての議論セグメントに新たな話題を提示している継続発言が存在する、つまり複数の話題が存在することが分かる。新たな話題を提示する継続発言として最も多かったのが、直前までの議論から派生した話題を提示する発言である。たとえば、直前までは「意味単位を分割できるか」という議論を行っていたが、「逆に意味単位を統合できるか」と別の視点から話題を切り出す発言がこれに該当する。「意味単位」という大きな視点から捉えると一貫した議論を行って

表 3.1: システムが生成した議論構造と正解データとの比較結果

ID	発言者数	総発言数	各評価項目別の発言数		
			(1)	(2)	(3)
1	6	14	1	2	2
2	5	13	1	0	3
3	5	13	1	0	3
4	4	10	2	0	0
5	4	13	2	0	2
6	4	8	1	0	2
7	4	7	1	0	1
8	4	6	1	1	1
9	5	17	1	0	1
10	4	11	1	0	2
11	4	11	1	0	0
12	3	5	0	0	0
13	5	16	6	0	0
14	5	13	1	1	3
15	4	14	1	0	1
16	7	11	2	0	0
17	5	9	1	0	1
18	5	8	0	0	1

いるが、より詳細な視点から捉えると別の話題について議論を行っている、という特徴がある。

現在使用している発言タイプだけでは柔軟な視点で議論を分割することはできない。一つの議論セグメントが長いほど閲覧に必要な時間が長くなるため、効率的な議論内容の閲覧を実現するためには、発言タイプと別の手法と組み合わせた、より細かいセグメント情報の取得が望ましい。たとえば、議論セグメントの親である導入発言と、それに続く継続発言の内容・意図の関連性に着目した方法が挙げられる。もし導入発言者が、自身の発言の内容・意図と関連のない継続発言の情報を入力できるならば、関連の弱い継続発言が集中している箇所は別の話題について議論していると推測できる。自動的にセグメント情報を抽出することは困難かもしれないが、人間が後付けでセグメント情報を入力する際の手掛かりとして十分な情報を得ることができるだろう。

(2) 親発言の異なる継続発言の傾向

表 3.1 から、親発言の異なる継続発言は全ての継続発言（181 発言）内の 2.2% に相当する 4 発言であることが分かる。このことから発言予約機能によって取得した

発言のリンク情報は妥当であると考えることができる。また、該当する発言を詳細に調べると、発言予約を入力するタイミングが遅くなったことが原因であることが分かった。発言予約機能は、発言予約を入れた時点で行われていた発言に対してリンク情報を付与するものであるが、発言予約を入れた時点で発言が行われていない場合は直前の発言に対してリンク情報を付与するようになっている（導入発言はこれに該当しない）。そのため、直前の発言よりも前に行われた発言に対して、リンク情報を付与することができない。この問題点の解決法としては、発言者が発言を行う際に対象となる発言を指定する手法が挙げられる（詳細については4.2.1項にて述べる）。

（3）複数リンクを持つ継続発言の傾向

表3.1から分かるように複数のリンクを持つ継続発言は数多く存在する。個々の発言内容を見ていくと「錯綜してきたので整理するけど」「君も言っていたけど」といったように、それまでの発言や議論を受けた意見やまとめを述べる発言が多かった。なお、複数のリンクを持つ継続発言が存在しない議論セグメントは質疑応答の繰り返しやシステムに対する要望を出し合うものが多かった。このことから「リンクの多い発言ほどその議論セグメントにおいて重要」「複数リンクを持つ継続発言が多いほどその議論セグメントは活発」といったように発言や議論セグメントの重要度や活性度を求める上で、複数のリンク情報を記録することは有効だと考えられる。

現状の発言タイプや発言予約機能だけでは、複数のリンク情報を取得することはできないが、上述の親発言の修正と同様にサブスクリーンとポインタリモコンを用いて参照したい発言を追加することで対処できるだろう。また、会議中に取得する手法だけでなく、会議後に入力する手法も存在する。たとえば、重要だと思われる発言を複数引用して、自分の意見やアイデアを記述する仕組みを実現することで、同時に引用されている発言間にリンク情報を付与できるだろう。

3.6.2 発言テキストの情報量に関する評価

会議コンテンツの閲覧や検索、要約などを実現する上で重要な情報となるのが発言テキストである。発言テキストを取得する方法として音声認識が挙げられるが、実環境における認識精度は高いとは言えず、仮に完全な書き起こしを作成することができたとしても、文法的に正しくない発言がそのまま書き起こされてしまうといった問題点がある。

Discussion Recorderでは上記の理由により、書記が発言を要約しながら入力するという方法を採用しているが、すべての内容を入力することが困難であるため、取得される発言テキストの情報量は減損していると考えられる。また、発言内容を忠実に書き起こそうとする書記もいれば、必要だと思われる内容だけ入力し、冗長と判断した内容は入力しない書記もいるように入力されるテキストの量や内容

は書記に依存すると考えられる．そこで書記が入力した発言テキストに関して次の項目について評価を行った．

- 書記による記述量の個人差
- 議論内容と記述量の関係

評価に用いたデータセットと評価方法

書記が入力した発言テキスト（以下，書記テキスト）と，ビデオを閲覧しながら書き起こした発言テキスト（以下，書き起こしテキスト）に含まれるキーワードの割合の比較を行った．

評価に用いるデータセットは，書記による記述量の個人差を考慮するため，著者が所属する研究室の大学生・大学院生6名が2007年度以降に書記を行った会議コンテンツの集合から12個を選出した．そのためにまず，6名の書記を1発言あたりの平均文字数によって3つのグループ（各2名ずつ）に分類した．ここで，各書記グループを平均文字数の多い順番にグループA，グループB，グループCと呼ぶことにする．そして，平均文字数に基づいて，それぞれの書記が担当した会議コンテンツから2個ずつ（各書記グループにつき計4個）選出した．ちなみに全ての会議コンテンツに含まれる発言の総数は661発言（1会議コンテンツあたり55.1発言）であり，議論セグメントの総数は137個（1会議コンテンツあたり11.4個）であった．

評価を行うために，まず正解データとなる書き起こしテキストの作成を行った．なお，この後の評価においてノイズになるため，間投詞や言い誤りの書き起こしは行わなかった．次に発言テキストにおけるキーワードの割合を求めるため，書記テキスト，書き起こしテキストのそれぞれに対して形態素解析ライブラリ Sen²を用いて形態素解析を行った．最後に形態素解析によって得られたキーワード候補の中からキーワードとなるものを選択した．キーワードの選択は「記述されている順番に見るだけでその発言の全ての内容が把握できる」という定義に基づいて行った．なお，発言テキストの書き起こしやキーワードの選択は，前述の6名の書記とは異なる大学生・大学院生3名に行ってもらった．この3名は事前に書き起こしやキーワード選択に関する注意事項を周知されているため，作成された書き起こしテキストや選択されたキーワードにはばらつきはないとみなす．

評価結果と考察

選出された会議コンテンツにおける発言テキストの総文字数は，書記テキストで41,946文字（1会議コンテンツあたり3,495文字），書き起こしテキストで188,816

²Sen . <https://sen.dev.java.net/>（2010年12月11日）

表 3.2: 書記テキスト, 書き起こしテキストにおける自立語, キーワード, 共通キーワードの数とその割合

Grp	ID	自立語, キーワード, 共通キーワードの数					自立語, キーワード, 共通キーワードの割合			
		①	②	③	④	⑤	②/④ (%)	②/① (%)	⑤/② (%)	⑤/④ (%)
A	1	8.2	5.0	17.7	8.1	3.7	61.1	60.8	75.1	46.5
	2	19.6	11.3	35.1	14.3	7.4	79.0	57.5	65.8	52.0
	3	7.5	4.6	15.7	8.3	3.9	55.7	61.2	83.8	46.6
	4	11.7	7.6	26.4	14.2	6.1	53.2	64.8	80.8	42.9
B	5	7.3	4.8	16.3	8.9	3.5	54.1	66.4	72.2	39.3
	6	5.2	3.1	11.9	5.0	2.2	61.3	59.0	71.2	43.9
	7	8.3	5.9	15.4	9.8	4.9	60.4	70.7	83.3	50.3
	8	5.8	3.8	11.9	7.0	3.2	53.3	64.6	84.8	45.2
C	9	4.7	3.1	13.2	5.7	2.5	54.9	66.6	80.5	43.9
	10	5.4	2.3	16.1	3.6	1.7	64.0	42.0	75.2	47.6
	11	3.7	2.9	10.4	6.5	2.1	44.3	77.1	73.1	33.1
	12	5.5	4.1	15.7	8.7	3.3	47.5	74.4	80.2	37.3
平均		7.8	4.9	17.2	8.3	3.7	57.4	63.7	77.2	44.1
標準偏差		4.3	2.5	7.0	3.3	1.7	8.9	9.1	5.9	5.4

文字 (1 会議コンテンツあたり 15,734 文字) であった。また, 形態素解析によって得られた全ての形態素数 (重複を含む) は書記テキストで 23,950 個 (1 会議コンテンツあたり 1,995 個), 書き起こしテキストで 105,843 個 (1 会議コンテンツあたり 8,820 個) であった。文字数や形態素数の観点から見ると, 全ての発言内容を書き起こすために必要な記述量は, リアルタイムに書記が入力する記述量の約 4 倍であることが分かった。このことから発言内容を全て書き起こすためには, かなりの時間を必要とするため, 会議中に行うことは困難であることが分かる。

書記テキスト, 書き起こしテキストにおける自立語, キーワード, そして書記テキスト, 書き起こしテキストのどちらにも出現するキーワード (以下, 共通キーワード) の数の集計結果を表 3.2 に示す。左から書記のグループ, 対象の会議コンテンツの ID, 書記テキストにおける 1 発言あたりの自立語の数 (①), キーワードの数 (②), 書き起こしテキストにおける 1 発言あたりの自立語の数 (③), キーワードの数 (④), 共通キーワードの数 (⑤), それぞれの値を用いた割合を表している (表中の丸番号は図 3.9 と対応)。また, 表中の①から⑤までの値は発言数

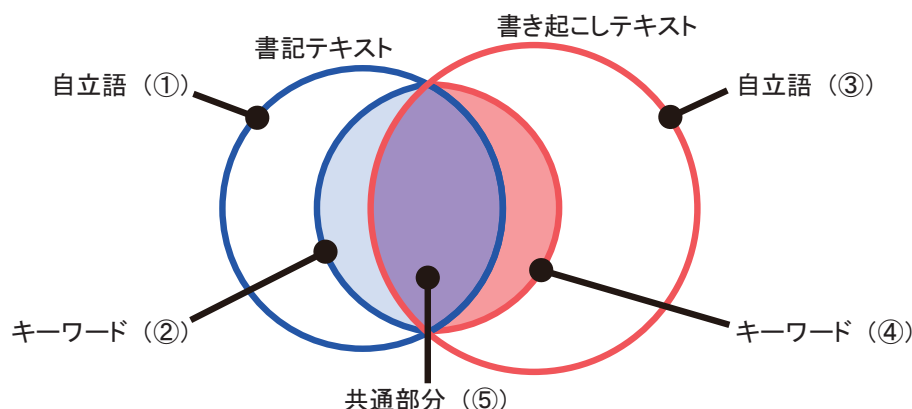


図 3.9: 書記テキスト，書き起こしテキストにおける自立語，キーワード，共通キーワードの包含関係

で正規化している。

なお，説明のため，書記テキスト内のキーワード数と書き起こしテキスト内のキーワード数の比率を示す②/④を，書記によってキーワードがどれだけ書き起こされたかを測る指標と捉え，書き起こし率と呼ぶことにする。また，書き起こしテキスト内のキーワードに対する共通キーワードの含有率を示す⑤/④を，書記がいかに発言内容を忠実に記述したかを測る指標と捉え再現率と呼ぶことにする。

再現率の平均が 44.1%であることから，発言内容のキーワードの半分以上が入力時に欠落していることが分かる。今回の評価では，キーワードを選択する基準を「記述されている順番に見ただけでその発言の全ての内容が把握できる」としたが，たとえば発表者が会議コンテンツを閲覧する際には，自身の研究を進める上で必要な発言の，必要な部分さえ記述されていれば十分だと考えられる。そのため実際に，発表者に対して自分が発表した会議コンテンツの書き起こしテキストの中で必要な発言，さらにはその中でも最低限記述されていてほしい内容に対してチェックをしてもらい，書記テキストと比較したところ，書記テキストだけで十分な内容が記述されていたことが確認された。これに対して，会議コンテンツや議論セグメント，発言の検索にはより多くのキーワードが含まれていることが望ましいだろう。このように利用目的によってキーワードの重要度は異なるため，書記の入力時に欠落した情報の中に，重要な情報がどれだけ含まれていたかを評価するためには，利用目的に応じてどのようなキーワードが必要であるかを踏まえた上で個別に検討する必要がある。

(1) 書記による記述量の個人差の検証

書記グループごとに書き起こし率（②/④），再現率（⑤/④），書記テキストにおける共通キーワードの割合（⑤/②），書記テキストにおけるキーワードの割合（②/①）の平均，標準偏差を比較した結果を図 3.10 に示す。

1 発言あたりの平均文字数によってグループ分けを行ったため，書き起こし率も

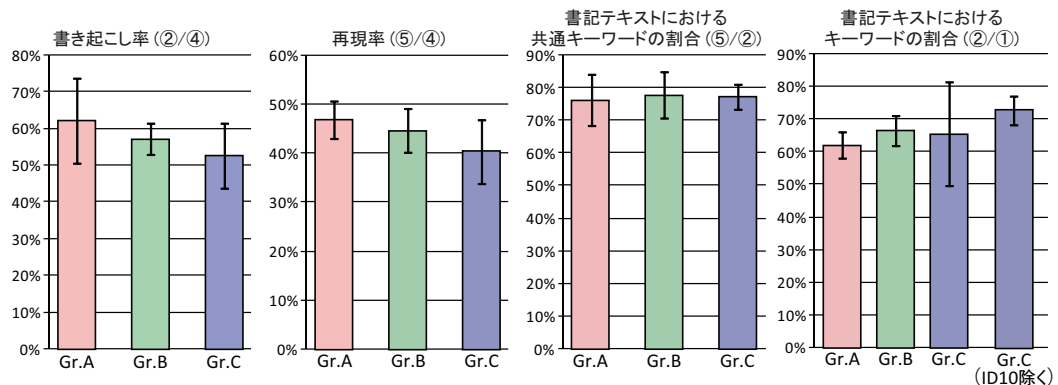


図 3.10: 書記による発言テキストの記述量の個人差

グループごとに差が存在することが分かる。また、再現率のグラフから、書き起こし率が高くなるにつれて発言中のキーワードがより多く書き起こされていることが分かる。その一方で、書記テキストにおける共通キーワードの割合を示すグラフを見ると、それほど個人差は見られないことが分かる。つまり、書記によって入力されるテキストの量とは関係なく、書記テキストにおけるキーワードの割合はほぼ一定であると推定される。このことから入力されるテキストの量は書記によって個人差があるが、その質自体に個人差はほとんど存在しないことが推測できる。また、書記テキストにおけるキーワードの割合のグラフ (②/①) では、グループ C の標準偏差が大きいですが、これは ID10 の書き起こしテキストにおけるキーワードの割合 (④/③) が他の会議コンテンツに比べて非常に小さい、つまり書記が入力すべき発言内容に含まれるキーワードそのものの数が少ないため、結果として書記テキストに含まれるキーワードの割合も小さくなったと考えられる。このグラフの一番右側の値は ID10 を除いた書記テキストにおけるキーワードの割合を表したものである。このグラフから平均文字数が少ない書記は多い書記に比べ、より簡潔に書記テキストを入力している可能性があることが分かる。つまり、平均文字数の多い書記は発言者の発言内容を逐一詳細に記述しようとしているのに対し、平均文字数の少ない書記は発言内容を要約しながら入力を行っていると考えられる。

(2) 議論内容と書記テキストの記述量の関係

一つの発表の中には質疑応答だけの議論やテーマに関する意見の出し合いのようにさまざまなタイプの議論が存在する。そして、質疑応答は意味が理解しやすく書記テキストが書きやすい、概念的な議論は理解しづらく書記テキストが記述しにくいといったように議論のタイプと書記テキストの記述量には関係があると考えられる。そのため、各会議コンテンツ内の議論セグメントに対して書き起こし率と再現率を求め、それぞれの値 X_i に対して以下の式によって基準値 SS_i を

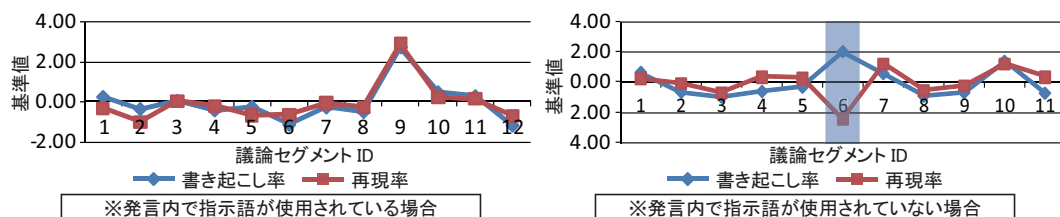


図 3.11: 書き起こし率と再現率の相関関係

算出し、グラフにプロットした（例を図 3.11 に示す）。なお、 $E[X]$ は X の平均、 $SD[X]$ は標準偏差を表している。

$$SS_i = \frac{X_i - E[X]}{SD[X]}$$

ほとんどの会議コンテンツでは、図 3.11 の左側のように書き起こし率と再現率はほぼ同じような曲線を描いており、書き起こし率と再現率には相関関係があると考えられる。しかし、図 3.11 右側のグラフにおける ID6 の議論セグメント（網かけ部分）のように書き起こし率と再現率が大きく異なる議論セグメントが存在することが分かった。このような議論セグメントを詳細に調べてみたところ、スライド内の図について言及している議論であることが分かった。このような議論では図を指し示しながら発言を行っているため、指示語を多用する傾向があった。そのため、書記が指示語の内容を補う単語を入力しているが、発言内容には直接含まれているわけではないため、再現率が低くなることが分かった。

(3) 発言に含まれる指示語の傾向

上記の結果のように発言内容に指示語が含まれている場合、その箇所だけを閲覧しても内容が把握できない上に、発言におけるキーワードとして利用することが困難なため、何らかの処置が必要になる。今回対象とした会議コンテンツ内の書き起こしテキストには指示語が平均で 28.0% 含まれていることが分かった。発言に含まれる指示語の傾向は大きく分けて (1) それまでの発言や議論を指し示すもの (2) スライド中のテキストを指し示すもの (3) スライド中の図形を指し示すもの、の 3 つとなることが分かった (1) は「その問題は…」といったようにそれまでの発言や議論を指し示す指示語であり、指示対象である発言は発言予約機能によるリンク付けが可能のため、リンク情報を用いた閲覧によって指示語の内容を確認できる。また (2) のようにスライド中のテキストを指し示す場合は、レーザーポインタなどを用いてスライドを指示することが多い。Discussion Recorder ではポインタリモコンを用いたスライドの指示やポインタの座標情報の取得、指示されたスライド中のテキストを抽出する手法を実現しており、スライド中のテキストが指示された際には、書記インタフェース上で指示されたテキストを補完する機能を追加することで (2) のような指示語の内容の入力を促すことができると考えられる。そして最後に (3) のようなフローチャートなどスライド中の図形

を示す指示語は、テキスト情報として記録することが困難な情報である。そのため、ポインタリモコンによって取得されたポインタ情報をスライドのサムネイル上で再生する仕組みを実現することによって発言内容の理解を促進できるだろう。

3.7 運用実績とユーザビリティに関する考察

著者が所属する研究室では、2003年度から約7年間にわたり Discussion Recorder を運用してきた。作成された会議コンテンツは年間約100件（録画された映像・音声は約150時間）にのぼる。継続的な運用の中で、Discussion Recorder を利用したために、議論にかかる時間が極端に長引いてしまうことや、書記以外の特定の参加者に特別な負担がかかるようなことは、経験的になかったため、今後も運用を継続していくことは可能だと考えている。

Discussion Recorder では、会議に関するメタデータの取得をシステム化しやすくするため、ポインタリモコンを用いた発言のタイムコードの入力や発言予約機能によって、参加者同士の発言のクロストークを排除する方法を採用している。この方法は議論の進行方法に対して制約を加えるものであり、自然な議論を妨げてしまう可能性がある。たとえば、ブレインストーミング形式の会議では、参加者同士の発言のクロストークを排除することによって、アイディアの創出を妨げてしまう可能性がある。同研究室では、ゼミを対象として会議コンテンツの作成を行っているため、参加者に対して「発表者である学生の研究活動にとって少しでも有益だと考える発言は記録し、雑談のように関係のない発言は記録しない」という基準を設けることで、できるだけ自然な議論が行えるようにしている。また、ちょっとしたあいづちや同意発言を行いたい場合は、あえて発言を行わず、ポインタリモコンのボタンを利用して発言に対する賛同を表明するなど、より簡易な操作を行うことができる。

また、発言を開始する際には手を挙げるようにポインタリモコンを掲げるなど、参加者ができるだけ自然にシステムを利用できるようにしている。しかし、システムの不具合や参加者のデバイス利用の不慣れなどによって、円滑に発言を開始できずに議論が一時的に停滞してしまうこともある。ただし、会議後に会議コンテンツを閲覧する際には、発言の開始・終了時間が記録されているため、このような停滞している時間をスキップして閲覧することができる点が特長である。

3.8 本章のまとめ

本章では、会議コンテンツを作成するための仕組みである Discussion Recorder について述べた。Discussion Recorder は、環境に設置されたセンサや参加者が所持するデバイスを用いて、映像・音声や発言テキスト、スライド情報の他に、発

言者情報や議論構造といったメタデータを組み合わせた会議コンテンツの作成を行う。

また、Discussion Recorder によって作成された会議コンテンツを分析することで、取得されたメタデータの評価を行った。具体的には、ポインタリモコンや発言予約機能によって取得される議論構造の妥当性、および書記が入力した発言テキストの情報量に関して評価を行った。その結果、以下のような結果が得られた。

- 議論構造の妥当性
 - － 導入・継続という発言タイプによって生成される議論セグメントの中には、詳細な視点からとらえるといくつかの話題が含まれる
 - － 発言予約機能によって付与される発言間のリンク情報の誤りは少ない
 - － それまでの発言や議論を受けた意見やまとめを述べる発言のように、複数のリンク情報を持つ発言が存在する
- 発言テキストの情報量
 - － 入力されるテキストの量は書記によって個人差があるが、その質自体に個人差はほとんど存在しない
 - － 書記は、発言に含まれる指示語の内容を補いながらテキスト入力を行う

次章では、Discussion Recorder によって作成される会議コンテンツを利用する仕組みについて述べる。

第4章 会議コンテンツの会議後および会議中における利用

4.1 会議コンテンツの検索・閲覧

Discussion Recorder によって、発言者や発言内容、発言時間など議論に関するさまざまなメタデータを取得できることを前章で述べた。そして、取得したメタデータを用いて効率的な議論内容の閲覧を実現することによって、議論内容のより深い理解につながり、知識活動の活性化を促すことが期待できる。

Discussion Recorder で作成される議事録は、テキスト主体の閲覧に有効なデータ構造を持っているが、議論内容の正確な把握には、映像・音声を含むマルチメディア情報を用いた閲覧のほうが有効であると考えられる。本研究では、テキスト主体の議事録に映像・音声情報やメタデータを組み合わせたものを会議コンテンツと呼んでいる。

マルチメディア情報には膨大な量の情報が含まれているため、閲覧目的に合わせた閲覧方法を提供する必要がある。会議コンテンツの閲覧目的は、重要な発言の再確認や自分が参加できなかった議論の内容把握など数多く存在する。このようにさまざまな閲覧目的を包括できるような仕組みを実現するために著者が所属する研究室では Discussion Browser^[35] と呼ばれる会議コンテンツを効率的に閲覧するためのシステムの研究・開発を行っている。

4.1.1 インタフェース XML の作成

Discussion Recorder で作成された議事録は、テキスト主体の閲覧に有効なデータ構造を持っている。しかし、ビデオ主体の閲覧を行うとき、ユーザが行う操作は時系列に基づいたものであることが推測される。つまり、ビデオによる議論内容の効率的な閲覧を行うためには、会議の構成要素ごとに構造化されたデータではなく、時系列ごとに構造化されたデータを作成することが望ましい。また、会議の内容を詳細に分析すると、発表者が資料を用いて自分の考えを述べている場面や議論が停滞してしまっている場面などいくつかの場面に分けられる。本研究では、ビデオ主体の閲覧を効率的に行う前処理としてインタフェース XML と呼ばれるデータの作成を行う。

```

<?xml version="1.0" encoding="Shift_JIS" ?>
- <interface id="tsuchida_060927">
+ <resource>
+   + <personList>
+   + <videoList>
+   + <slide src="http://www3.nagao.nuie.nagoya-
+     u.ac.jp/minute/slide/tsuchida_060927/tsuchida_seminar060927.ppt">
+   + <statements>
+ </resource>
- <timeline end="3095" start="0">
+   + <ps end="8" id="PS1" start="0">
+   + <ds end="164" id="DS1" speakers_num="2" start="8" statements_num="6">
+   + <ps end="660" id="PS2" start="164">
+   + <ds end="790" id="DS2" speakers_num="2" start="660" statements_num="2">
+   + <ps end="886" id="PS3" start="790">
+   + <ds end="1202" id="DS3" speakers_num="4" start="886" statements_num="9">
+   + <ds end="1462" id="DS4" speakers_num="2" start="1243" statements_num="3">
+   + <ps end="1637" id="PS4" start="1462">
+   + <ds end="2096" id="DS5" speakers_num="3" start="1637" statements_num="8">
+   + <ps end="2191" id="PS5" start="2096">
+   + <ds end="2241" id="DS6" speakers_num="1" start="2191" statements_num="1">
+   + <ps end="2462" id="PS6" start="2241">
+   + <ds end="2741" id="DS7" speakers_num="3" start="2462" statements_num="7">
+   + <ps end="3095" id="PS7" start="2741">
+ </timeline>
</interface>

```

構成要素に関する情報

時間情報に関する情報

図 4.1: インタフェース XML

インタフェース XML は図 4.1 のような構造を持っており、ビデオや発言、参加者やスライドなど議論の構成要素に関する情報とビデオの再生に関する時間情報の 2 つに大きく分けられる。また、インタフェース XML のルートノードには議論を特定するための ID が記述されている。

議論の構成要素に関する情報はルートノードの下にある resource 要素に記述されている。resource 要素は videoList 要素、slide 要素、personList 要素、statements 要素の 4 つから構成されている。videoList 要素には発表者 (presenter) やスクリーン (screen) といったビデオのタイプや URL、ID が記述されている。また、slide 要素は議論に利用されたスライドの情報が記述されており、スライド内のページタイトルやページテキスト、アニメーション情報がこれに該当する。発表者や書記といった参加者に関する情報は personList 要素に記録される。また、議論中に行われた発言に関する情報は statements 要素に記述される。具体的には、発言者の ID、発言のタイプ、発言テキスト、その発言に対するスタンス (賛成、反対) の数がこれに含まれる。

関連する議論や発言等をまとめて視聴することで、議論内容の深い理解に繋がることが予想される。しかし、これらの情報を自動的に抽出することは困難である。本研究では、導入・継続という発言タイプによって抽出される発言の集合に対応した時間区間を議論セグメント、それ以外の時間区間を発表セグメントと定義する。この 2 つのセグメントの時間情報は timeline 要素に記述される。timeline 要

素の下には、議論セグメントに関する情報を表す ds 要素と発表セグメントに関する情報を表す ps 要素があり、それぞれの要素にはセグメントの ID とセグメントの再生開始時間、終了時間が属性として記録されている。また、議論セグメントを表す ds 要素にはそのセグメントに含まれる発言の数や発言者の数も属性として記録されている。そして、それぞれのセグメントはアイテムと呼ばれる要素から構成されている。議論セグメントにおけるアイテム (議論アイテム) は発言の時間区間であり、発表セグメントにおけるアイテム (発表アイテム) はスライドのページの切り替えやアニメーション表示の切り替え、デモなど発表者の操作によって切り分けられる時間区間である。

ds 要素の子要素である di 要素は、議論アイテムの情報の発言を表している。議論アイテムに記述される情報を以下に示す。

id 議論アイテムの ID

start 議論アイテムの再生開始時間

end 議論アイテムの再生終了時間

ref 議論アイテムに対応する発言の ID

disp 対応する発言が行われていたときに表示されるスライドの ID

prev 一つ前の議論アイテムの ID (複数可)

next 一つ後の議論アイテムの ID (複数可)

pi 対応する発言が開始した時間に対応する発表アイテムの ID

各 di 要素には開始・終了時間や議論アイテムの ID、発言タイプのほかに、対応する発言の ID やその発言時に表示されていたスライドの ID などの情報が記録される。また prev・next 属性には、直前・直後の議論アイテムの ID が記述される。この情報を用いて閲覧時に議論アイテム間の移動を行うことで、発表内容を閲覧せずに議論内容だけを閲覧できる。この議論アイテム間の関係は、「導入」「継続」という発言タイプから暗黙的に取得されるだけでなく、書記が専用ツールで付与したリンク情報も考慮されるため、prev・next 属性に記述される ID は複数指定が可能である。また、発表の内容を受けて議論を開始することが多いため、各 ds 要素内で先頭に現れる di 要素には、直前の発表アイテムの ID が記述されている。

発表セグメントを表す ps 要素の子要素には発表アイテムを表す pi 要素がある。pi 要素に記述される情報を以下に示す。

id 発表アイテムの ID

start 発表アイテムの再生開始時間

end 発表アイテムの再生終了時間

ref 発表アイテムに対応するスライドやアニメーションの ID

disp そのときに表示されるスライドの ID

prev 一つ前の議論アイテムの ID (複数可)

next 一つ後の議論アイテムの ID (複数可)

di 発表アイテムから開始した発言に対応する議論アイテムの ID (複数可)

議論アイテムと同様に、発表アイテムにはアイテムの ID やビデオの再生開始時間・終了時間が記述される。また、先頭の議論アイテムに対応する発表アイテムの ID が記述されるのに対して、同じように発表セグメントの最終発表アイテムには、そこから派生した議論アイテムの ID が di 属性に記述される。発表者がスライド操作を行う前に異なる話題で議論を始めることがあるため、di 属性に記述される ID は複数指定が可能である。

4.1.2 システム構成

Discussion Browser で会議コンテンツを閲覧するときは、まず、システムのトップページに表示される会議コンテンツの一覧を見て、その中から閲覧したい会議コンテンツを選択する。システムは要求された会議コンテンツをデータベースから取得し、閲覧インタフェースを提示する。

さまざまな場面で起こる会議コンテンツの閲覧要求に対応するために、Discussion Browser は Web サーバ・クライアント型のシステムとして構築されており、閲覧者は特別なアプリケーションをインストールする必要は無い。システム全体の構成を図 4.2 に示す。

会議コンテンツ閲覧サーバ(以下、閲覧サーバ)は、インタフェース XML の生成・取得と、閲覧インタフェースの提示を担当する。会議終了後、議事録やリンク情報の変更があるたびに、議事録データベースとアノテーションデータベースのデータを統合してインタフェース XML を自動的に生成し、インタフェース XML データベースに保存する。また、ユーザが会議コンテンツを選択すると、会議コンテンツの閲覧要求が閲覧サーバに送信される。閲覧サーバは要求されたコンテンツに該当するインタフェース XML をデータベースから取得し、後述する 5 つのコンポーネントから構成される閲覧インタフェースをユーザに提示する。

ユーザは目的に応じて、インタフェース内のコンポーネントを操作することで会議コンテンツの閲覧を行う。会議の様子を撮影したビデオは、ストリーミングサーバを通じてアクセスされる。

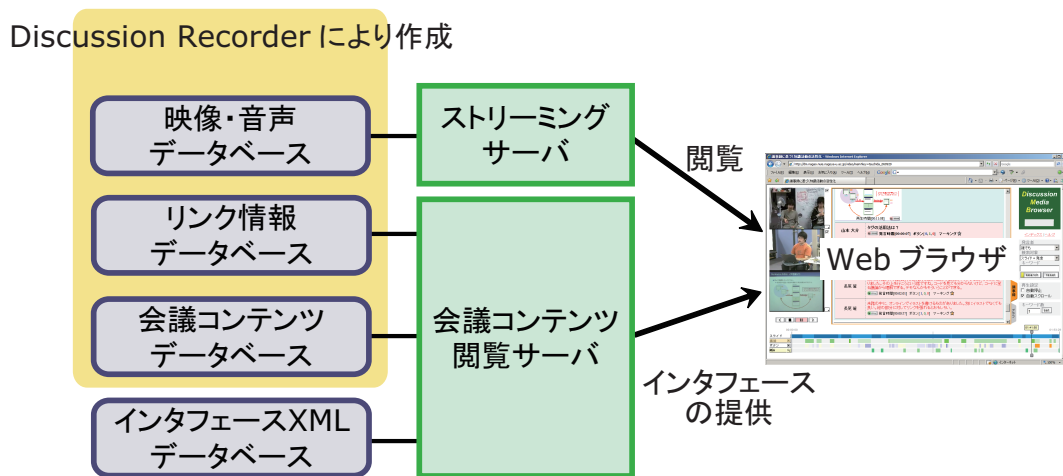


図 4.2: Discussion Browser のシステム構成図

4.1.3 インタフェース

Discussion Browser の全体を図 4.3 に示す。

Discussion Browser は以下に示すコンポーネントから構成されている。

- ビデオビュー
- 層状シークバー
- 議論内容詳細ビュー
- 検索テーブル
- 設定テーブル

これらのコンポーネントがそれぞれ相互に連携しながら動作することによって議論内容の効率的な閲覧を実現している。以下で、それぞれのコンポーネントの詳細について述べる。

ビデオビュー

議論中の様子を取得するためのビデオには、参加者ビデオ、発表者ビデオ、スクリーンビデオの 3 種類がある。ビデオビューはこれら 3 つのビデオの同時閲覧を実現している (図 4.4)。また、閲覧する環境や閲覧者の要求に合わせてビデオの表示・非表示を切り替えたり、縮小された映像では確認できないスクリーンの文字などを確認するためにビデオの拡大を行ったりできる。そして、再生や停止、スキップなどのビデオ操作を行うことも可能である。

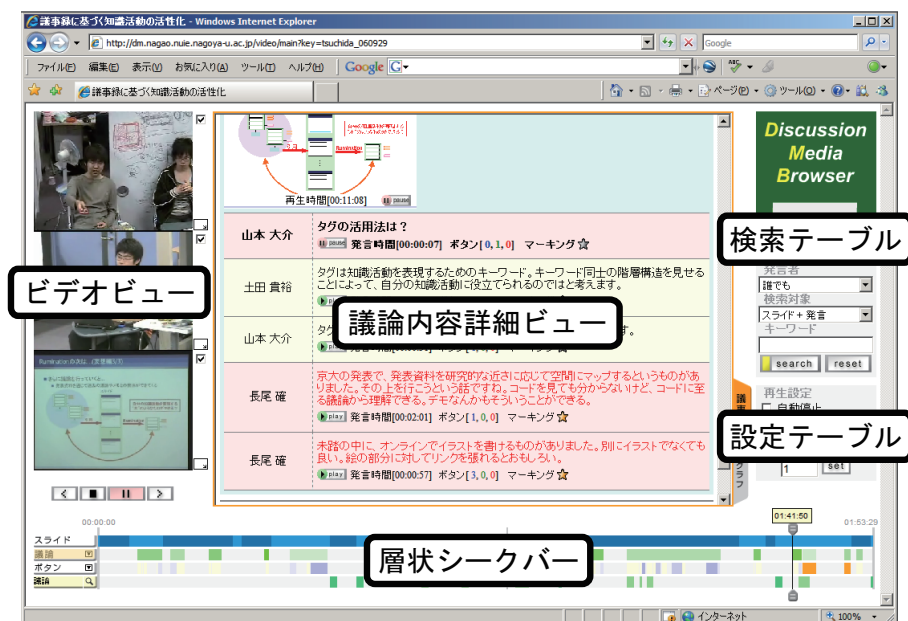


図 4.3: Discussion Browser

このビデオビューは後述する層状シークバーや議論内容詳細ビュー上の操作と連動しており、発言単位やスライド操作単位でのスキップ再生を行うことができる。逆にビデオビューにおけるビデオの再生時間に応じて層状シークバー内のスライダーや議論内容詳細ビューの表示が変化することで、議論全体の中のどの時点を読覧しているのかを確認できる。

層状シークバー

Discussion Recorder で取得したメタデータは、層状シークバーと呼ばれる図 4.5 に示すコンポーネント内のタイムライン上に表現され、議論を俯瞰する機能を提供する。層状シークバーはスライダーと複数のバーによって構成される。

スライダーの上部には現在再生されているビデオの時間が表示されており、再生時間に応じてタイムライン上を移動する。また、スライダーをドラッグすると、ビデオビューにそれぞれのビデオのサムネイル画像が表示され、議論中の任意の位置にアクセスできる。ドラッグが終了するとその時点からビデオの再生が開始される。

層状シークバーでは、Discussion Recorder で取得したメタデータごとにタイムライン上にバーを作成することによって、さまざまな視点から議論全体を俯瞰することができる。スライダーは発表者のスライド操作を表現している。色の変わり目がスライドの切り替えを表しており、それぞれの箇所にマウスカーソルを合

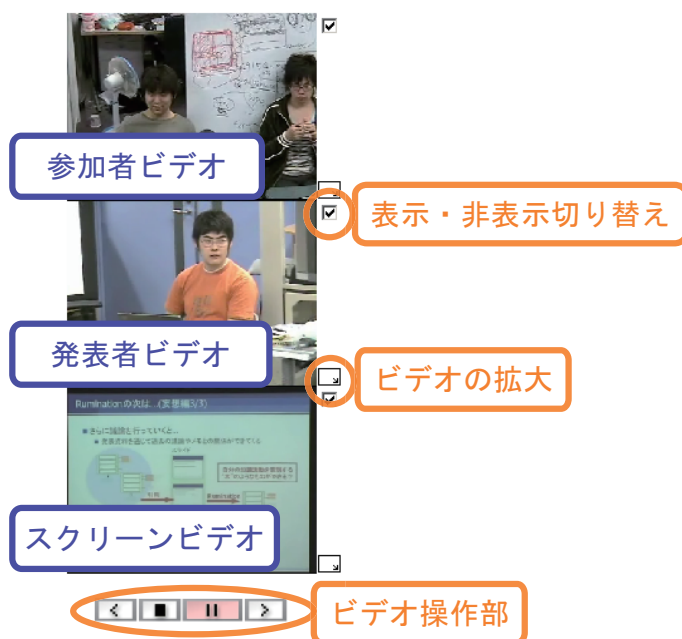


図 4.4: ビデオビュー

わせるとそのときに表示されていたスライドのタイトルを確認できる。また、クリックをすることによって、該当するスライドの開始時間からビデオの再生が開始される。

発表中に行われた議論の情報は議論バーに表示される。議論バーは議論セグメントを単位とするアイテムから構成されており、それぞれの議論セグメントにおける単位時間当たりの発言数によって表示される色が異なる。アイテムをクリックすると、その議論セグメント内の導入発言からビデオの再生を開始する。また、それぞれのアイテムにマウスカーソルを合わせると図 4.6 のように該当する議論セグメントの概要がポップアップウィンドウに表示される。議論セグメントの概要には、その議論セグメントにおける発言者のリスト、発言者数、発言数に加え、発言に含まれるキーワードを表示している。これらの情報を手がかりに、その議論セグメントではどのような内容の議論が行われていたかを把握できる。

ボタンバーは、発言に対して押された賛成/反対の数を表示している。バーをクリックすると該当する発言の再生を開始する。賛成ボタンを多く押された発言ほど青く、反対ボタンを多く押された発言ほど赤く表示される。また、その発言に対して押された賛成・反対ボタンの情報全てを確認することも可能である。また、5章で後述するマーキング情報もこのバーで確認できる。その際、ある発言に対して押された賛成・反対の情報とマーキング情報が重なってしまう可能性がある。ボタンバーの情報は、どの発言を閲覧すればいいのかを判断するための手がかりであるという観点から捉えると、賛成・反対の情報よりも個人が「重要である」「役

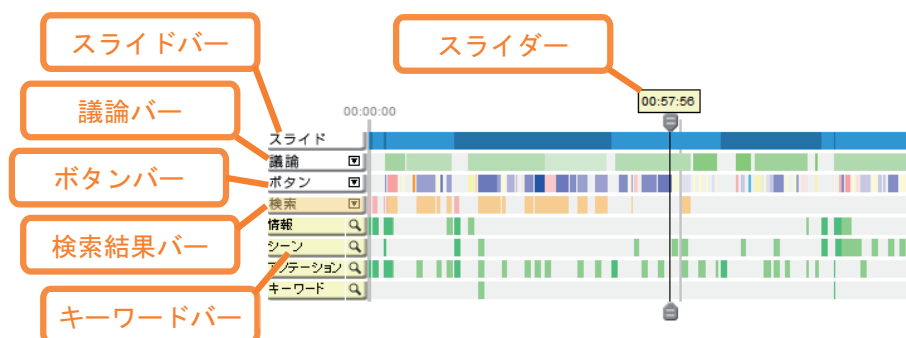


図 4.5: 層状シークバー

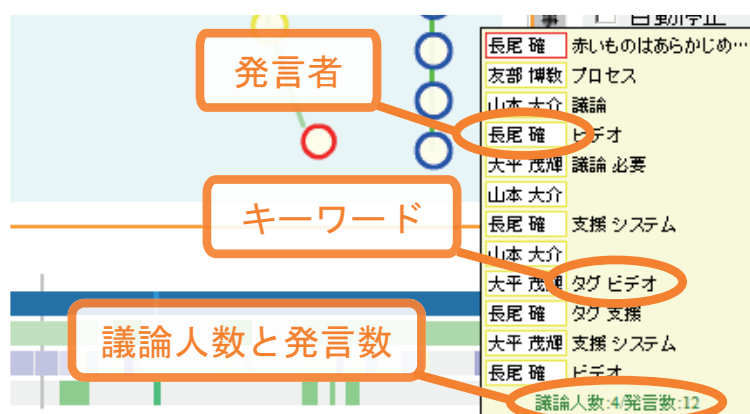


図 4.6: 議論セグメントの概要表示

に立つ」と判断した情報をより重視すべきである。そのため、賛成・反対の情報とマーキング情報が重なったときはマーキング情報を優先的に表示する。

検索テーブル上で行われた検索の結果は検索結果バーに表示される。Discussion Browser における検索は、スライドや発言に対して行うことができ、スライドに関する検索結果はピンク色で、発言に関する検索結果はオレンジ色で表示される。結果をクリックすると、該当箇所がスライドの検索結果ならばそのスライドの開始時間から、発言の検索結果ならばその発言の開始時間からビデオの再生を開始する。

また、発表や議論の内容を巨視的に把握するために、層状シークバーにはキーワードバーが用意されている。キーワードバーは、発表・議論に現れたテキスト情報を形態素解析することによって得られたキーワード¹を表示するバーである。そ

¹現在、キーワードの抽出は TF-IDF 法を用いて行っているが、今後は Discussion Browser の長期的な運用を通じて、議論構造や発言に対する賛成・反対・マーキング情報、スライドテキストも併せた、よりの確なキーワードの抽出アルゴリズムを考えていくことが望まれる。

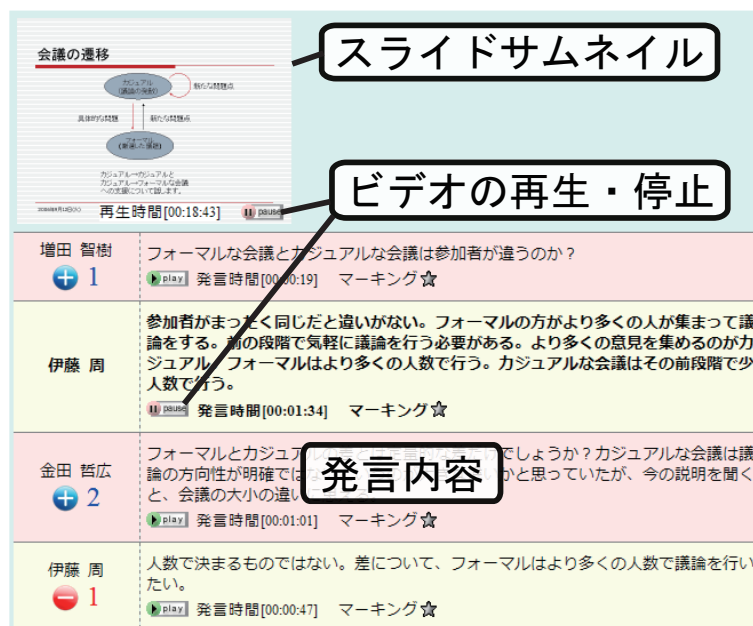


図 4.7: 議事録ビュー

のキーワードが現れている箇所がスライドならば濃い緑色、発言ならば薄い緑色によって表現される。検索結果バーと同様にクリックしたアイテムに該当する箇所がスライドか発言かによって再生を開始する時間が異なる。また、キーワードバーは設定テーブルによって、表示する数を変更できる。

議論内容詳細ビュー

層状シークバーによってさまざまな視点から議論全体の俯瞰を行った後は議論内容詳細ビューによってその詳細を閲覧できる。議論内容詳細ビューは、議事録ビュー (図 4.7) とグラフビュー (図 4.8) の 2 種類のビューから構成されており、右側のタブによってどちらかのビューを選択することで、閲覧したい内容に合わせた情報提示を実現する。

議論の内容をテキスト中心に閲覧するには議事録ビューを利用する。議事録ビューは、発表に用いられたスライドのサムネイル画像や個々の発言内容を表すテキスト情報、メタデータから構成されている。スライドのサムネイル画像をクリックすることによって、別ウィンドウに拡大表示され、より詳細な発表内容を確認できる。そして、個々の発言内容には書記が入力したテキストに加え、発言者 ID、発言時間、その発言に対して押された賛成/反対の数、マーキングの情報が含まれる。ユーザは、これらの情報を元にビデオの閲覧を行う。また、ビデオの再生時間に対応するスライドや発言はハイライト表示され、現在閲覧している箇所を容易に

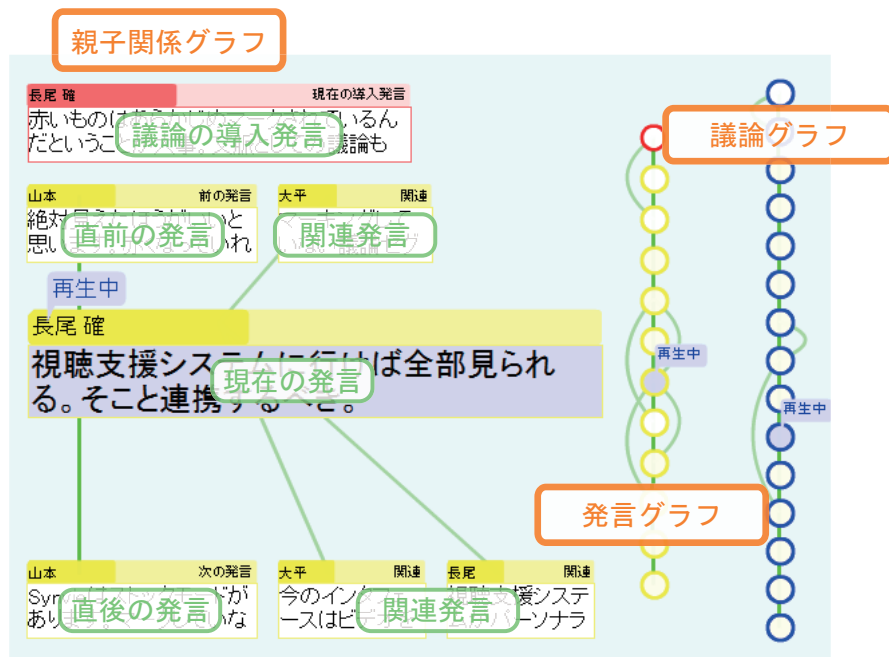


図 4.8: グラフビュー

把握できる．さらに，設定テーブルで自動スクロール機能を有効にしておくことによって，再生しているスライドや発言が切り替わったとき，議事録ビューは該当するスライドや発言に自動にスクロールし，フォーカスを当てる．

Discussion Recorder で取得するメタデータに，導入・継続という発言タイプによって付けられる発言の親子関係がある．議事録ビューでは，個々の発言を時系列順に表示しているため，議論を順番に閲覧することには適しているが，発言タイプや発言予約機能によって生成された発言間の関係を把握することは困難である．この関係を把握しやすいグラフ形式で提示するのがグラフビューである．グラフビューは，親子関係グラフ，発言グラフ，議論グラフという3つのグラフによって構成されている²．

親子関係グラフは，ある発言を中心とする発言間の関係を表現したグラフである．グラフの中央にある発言の上下には，その発言の派生元である親発言，派生先である子発言が表示されている．最上部には中央の発言が含まれている議論セグメントの導入発言が表示されているが，もし中央の発言が導入発言であるときは一つ前の議論セグメントの導入発言が表示される．中央の発言をクリックすることによって，その発言のビデオを閲覧できる．また，親発言や子発言，導入発言をクリックすれば，その発言が中央に移動し，その発言の親子関係が新たに現

²2010年12月現在，議事録ビューとグラフビューは，発言タイプや発言予約機能によって生成された議論構造を木構造として表示するビューとして統合されている．

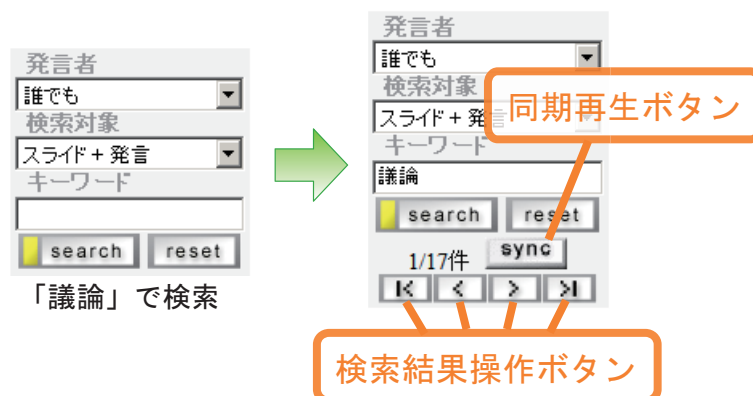


図 4.9: 検索の例

れる．これにより，発言間の関係を容易に把握することができ，テキストだけでは理解することが困難な文脈情報の理解を促すことができる．

検索テーブル

検索テーブルは，あいまいな手がかりをもとに閲覧者とスライドや発言を結びつける役割を果たす．同時に，検索結果に対してインタラクティブな操作を行うことによって，閲覧目的を明確にし，より具体的な閲覧対象へと閲覧者を導く手助けを行う．

検索テーブルでは，検索を行う対象（スライドか発言，もしくは両方），その対象が発言ならば発言者名，テキストを入力することによって検索を行う．検索例を図 4.9 に示す．検索を行うと該当した件数が検索テーブルの下部に表示される．また，議事録ビューや層状シークバー内の検索結果バーの該当箇所がハイライトされる．検索テーブル下部に表示された 4 つのボタンによって，検索で該当した箇所を移動できる．また，同期再生ボタンを押した状態で検索結果間の移動を行ったときは，移動した先のスライドや発言のビデオ再生を開始する．

閲覧者は，検索テーブルと層状シークバー・議事録ビューとの操作を繰り返すことによって会議コンテンツの鳥瞰的・局所的な閲覧を適切に行うことができる．従来の検索に比べ，よりインタラクティブな操作を行うことで効率的に閲覧目的を達成し，内容の理解を深められることが期待される．

設定テーブル

設定テーブルでは，ビデオの再生に関する設定と層状シークバー内のキーワードに関する設定を行うことができる．ある発言の再生が終了したときに「自

動停止」が有効になっていれば、自動的にビデオの再生が停止される。また「自動スクロール」が有効になっているときは、再生している発言やスライドに合わせて議事録ビュー、もしくはグラフビューのハイライト箇所が自動的にスクロールされる。これらの設定をユーザが行うことによって、スムーズにビデオを閲覧するという目的と時間をかけて内容を閲覧するという目的の両方を満たすことができる。

「キーワード数」というテキストボックスに0から20までの数を入力することによって、層状シークバー内のキーワードバーの数を変更できる。キーワードバーの数を増やすことによって発表・議論内容の概要の把握を、数を減らすことによって議事録ビューやグラフビューなどの表示領域が拡大され、詳細な議論内容の把握を促すことが期待される。

4.1.4 会議コンテンツの利用状況に関する調査

本章では、これまで Discussion Recorder で作成された会議コンテンツの検索・一覧を行うためのアプリケーションである Discussion Browser について述べた。Discussion Browser の実現により、どんな時でも過去の議論内容を参照できるようになり、知識活動を進める中で積極的に利用されることが期待される。そこで本項では、会議コンテンツの利用状況を確認するために行ったアンケート調査の詳細について述べる。

調査方法

本調査における対象は男性7名であり、被験者らには2006年4月14日から2007年1月19日までの約10ヶ月間、Discussion Recorder を用いて自身の研究活動に関する発表・議論を数回行ってもらった。

その後、それぞれの被験者に対して、自身の発表で行われた議論の一覧を配布した。この一覧は Discussion Recorder で取得した議論セグメントの一覧である。そして、日付の古い順に全ての議論セグメントを見直してもらい、それぞれの議論セグメントが、以下に示す5つのカテゴリのどれに該当するかを判定・回答してもらった。

- A 現在その議論の内容を覚えており、なおかつすでにその内容に基づいた活動を行った（議論後に「その内容については作業を行わない」と判断したものも含める）
- B 現在その議論の内容を覚えているが、いまだにその内容に基づいた活動を行っていない

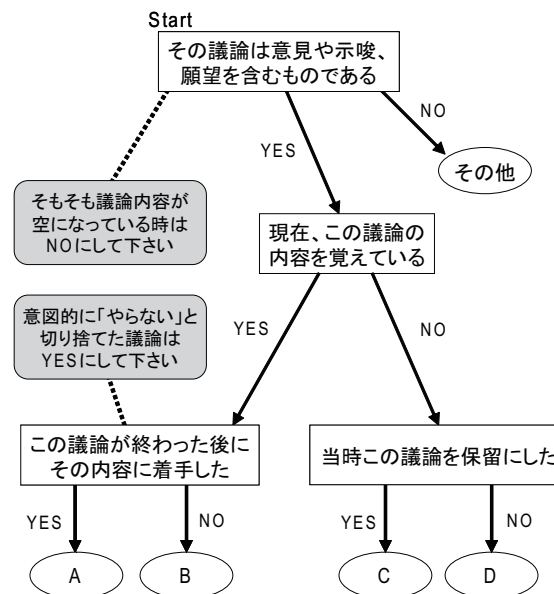


図 4.10: アンケート調査に用いた議論の判定木

C その議論の直後で、意識的に保留 (今すぐ取り組まなければならないものではないと判断した) にしたため、現在はその内容を忘れてしまっている

D 議論の直後でも全く意識せず、現在もその内容を忘れてしまっている

その他 質疑応答で終始していたり、その議論の中で問題が解決していたりする (書記の入力ミスといった議事録の不具合も含める)

図 4.10 に議論の判定に使用した条件木を、図 4.11 に被験者に配布した議論セグメントの一覧の一部を示す。

結果と考察

被験者の回答から得られた結果を表 4.1 に示す。この表では、期間を 1ヶ月ごとに分け、それぞれの月で行われた議論に対する判定結果をカテゴリごとに分類したものである。たとえば表 4.1 の「2006 年 04 月」「該当する議論の数 (D)」の箇所は、2006 年 4 月に行われた全ての議論セグメント 31 個のうち、12 個がカテゴリ D に分類されたことを表している。そして右側に、その月に行われた議論セグメント全体に対する割合を示す。さらに、期間を前期 (2006 年 4 月から 2006 年 8 月まで) と後期 (2006 年 9 月から 2007 年 1 月まで) の 2 つに分けた時の結果を表 4.1 の下部に示す。2006 年 8 月と 2007 年 1 月の議論セグメント数が少ないのは夏期休

1・DRIP モデルにおける発表者による有益な議論の促進
[2006-04-26・tsuchida] (11人)

○議論をスムーズに進めるには有意義だが、新たな情報発見などの応用はあるのか。
○フィードバックで自分の研究が発展していく。応用はどうか。
○計算機を使い議論を活性化させている。計算機の役割は何か。
○知識活動支援システムにおいて議題の設定をどのようにするかの話をする。DRIPの investigation においてできること。そこでシステムが提案してくれる。

この議論に対する判定は？(A・B・C・D・その他)
これより前に議論で同じ内容のものがあればその番号()

図 4.11: アンケート調査に用いたアンケート用紙の一部

暇と冬期休暇が重なったためである。また、被験者一人当たりの平均発表回数は 9.57 回であり、1 回の発表における議論セグメントの平均数は 9.84 であった。

まず、被験者全体における傾向について分析した。なお、説明のため、時点 i (2006 年 4 月から 2007 年 1 月まで順に $i = 0, 1, \dots, 9$ とする) において、カテゴリ A, カテゴリ B, カテゴリ C, カテゴリ D に分類された議論セグメントの割合をそれぞれ A_i, B_i, C_i, D_i とおく。たとえば、2006 年 8 月 ($i = 4$) においてカテゴリ A に分類された議論セグメント数の割合は 43.8% であるため、 $A_4 = 43.8\%$ である。

図 4.12 に各月における $A_i + B_i$ と $C_i + D_i$ の割合の推移を示す。カテゴリ A, もしくはカテゴリ B に分類された議論セグメントは、実験を行った時点でその内容を覚えているものであり、カテゴリ C, もしくはカテゴリ D に分類されたものはその内容を忘れていていることを表している。つまり、図 4.12 は時間の経過と共に議論内容がどれだけ忘失されていくかを表している。このグラフにおいて、より以前に行われた議論ほど、内容を忘れてしまっている議論セグメントの数が多いという傾向が分かる。つまり、議論が終わってから時間が経過すればするほど議論内容を忘れてしまうことが確認できる。

また、図 4.13 に各月における $A_i + C_i$ と $B_i + D_i$ の割合の推移を示す。カテゴリ B は議論内容を覚えていながらいまだに着手していない議論を表しており、カテゴリ D はこれまで全く注目されていない議論を表している。そのため、いずれかのカテゴリに含まれる議論は、知識活動において有効に利用されていない議論とみなすことができ、図 4.13 の $A_i + C_i$ は、知識活動を行う上で有効に利用されている議論セグメント数の推移を表していると考えられる。2006 年 4 月だけ $B_i + D_i$ の数が他の月に比べ大きな値をとっているが、全体的に見ればその値に変化はないとみなすことができる。つまり、知識活動において有効に利用される議論の数は時間の経過には依存しない傾向がある。

続けて被験者ごとの傾向について分析を行った。まず被験者ごとに内容を覚え

表 4.1: 被験者の回答から得られた議論セグメントの判定結果

発表の行われた月	該当する議論の数					全体に対する割合			
	A	B	C	D	計	A	B	C	D
2006 年 04 月	2	6	11	12	31	6.5%	19.4%	35.5%	38.7%
2006 年 05 月	21	17	19	4	61	34.4%	27.9%	31.1%	6.6%
2006 年 06 月	11	2	7	4	24	45.8%	8.3%	29.2%	16.7%
2006 年 07 月	19	10	13	5	47	40.4%	21.3%	27.7%	10.6%
2006 年 08 月	7	3	3	3	16	43.8%	18.8%	18.8%	18.8%
2006 年 09 月	13	1	3	4	21	61.9%	4.8%	14.3%	19.0%
2006 年 10 月	27	3	11	16	57	47.4%	5.3%	19.3%	28.1%
2006 年 11 月	29	4	7	6	46	63.0%	8.7 %	15.2%	13.0%
2006 年 12 月	24	4	4	3	35	68.6%	11.4%	11.4%	8.6%
2007 年 01 月	9	4	0	0	13	69.2%	30.8%	0.0%	0.0%
前期	60	38	53	28	179	33.5%	21.2%	29.6%	15.6%
後期	102	16	25	29	172	59.3%	9.3%	14.5%	16.9%

ている議論セグメント数の推移を調べたが、共通点を見出すことはできなかった。次に被験者 7 名のうち 5 名に対して、カテゴリ A、もしくはカテゴリ C に含まれる議論セグメント数を前期・後期の順に並べたグラフを図 4.14 に示す。被験者 7 名のうち、1 名は「その他」に含まれる議論セグメントの数が多く、前期に対して後期の有意セグメント数が非常に少ないためノイズとして除外した。そして、もう 1 名は前期と後期の間で研究テーマの大きな変更があったため除外した。また、被験者によって発表を全く行っていない月が存在するため、各月ではなく前期・後期という分け方を採用している。前期に対して、後期は議論が終了してから経過した時間が短いにも関わらず、有効に利用されている議論セグメント数はあまり増加していないことがこのグラフから分かる。

実験の結果から得られた傾向をまとめると以下のようになる。

- 被験者全体の傾向として、時間が経過するほど内容を忘れてしまった議論セグメントの数は増える
- 時間の経過に関わらず知識活動の中で有効に利用される議論セグメントの数は変わらない

前者の傾向から、再認フェーズにおいてアノテーションを付与する作業は、議論が終了した直後に行うことが望ましいことが分かる。また、後者の傾向について、被験者全体、被験者ごとに考察を行った。表 4.1 において、カテゴリ D に含

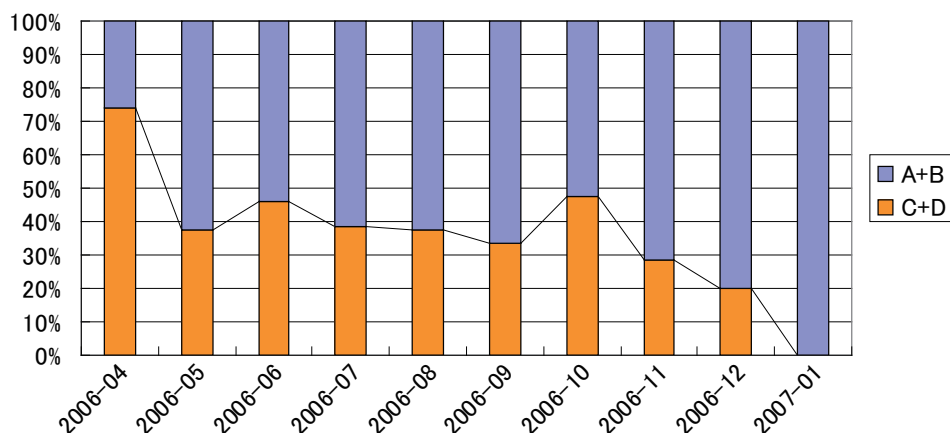


図 4.12: 内容を覚えている議論セグメント数の推移

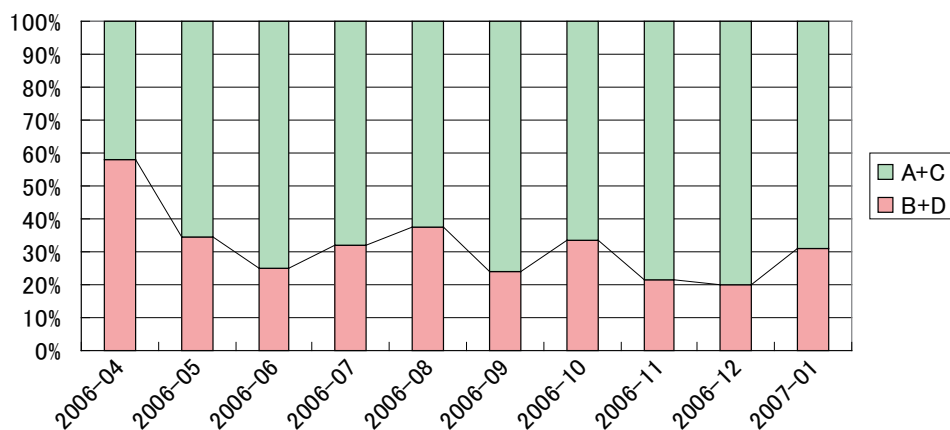


図 4.13: 有効に利用された議論セグメント数の推移

まれる議論セグメントの割合を、前期と後期で比べてみると大きく変化していないことが分かる。つまり、被験者全体において有効に利用された議論セグメント数が改善されない原因は、カテゴリ D に分類されるような議論の存在にあると考えられる。また、被験者ごとにカテゴリ B、カテゴリ D の割合を分析したところ、被験者ごとにその割合はばらつきがあり、傾向を見出すことはできなかった。これは発表テーマの内容や発表回数などさまざまな要因が被験者ごとによって異なるためだと考えられる。カテゴリ B に含まれる議論を改善するためには、被験者のモチベーション自体を改善する必要があるため、システムによって支援することは困難である。しかし、カテゴリ D に含まれる議論に関しては、再認フェーズにおいてアノテーションを付与することによって少なからず改善できる可能性がある。

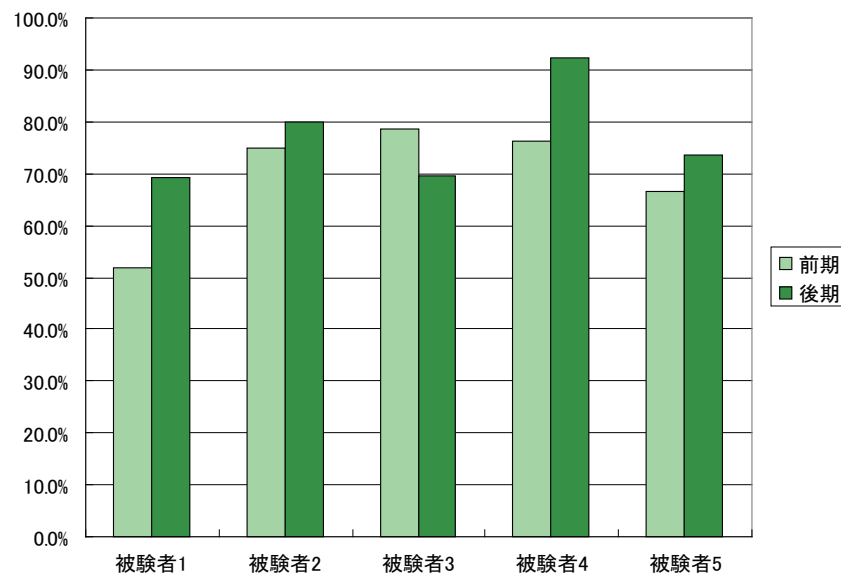


図 4.14: 被験者ごとの有効に利用された議論セグメント数の推移

4.2 会議コンテンツの利用に基づく議論の円滑化

前節では、Discussion Recorder によって作成された会議コンテンツの検索・閲覧を行うためのインタフェースについて述べた。本節では、会議コンテンツの作成・検索・閲覧だけでなく、議論そのものを活性化させるための枠組みについて検討を行う。では、そもそも議論を活性化するというとはどういうことだろうか。本研究において議論の活性化とはさまざまな理由で停滞している議論を以下のように変えることであると考える。

- 参加者が議論の発散・収束のバランスを適切に保っている
- 参加者全員が偏りなく、積極的に発言を行っている

まず、1 番目の項目については、システムが参加者に対して現在行われている議論の内容や進行状況を提示することが有効である。本論文では、Discussion Recorder で取得した議論構造に関する情報を効果的に可視化する仕組みを提案する。議論構造の可視化には以下のようなメリットがある。一つは、現在行われている議論内容を簡単に把握することができ、発言者が発言内容を整理できる点である。もう一つは、誰が過去のどの発言を受けて発言しようとしているのかを理解することによって、発言権を他者に譲るなど、議論を効率的に進められるようなファシリテーションを参加者に促すことができる点である。また、重要な話題であるにもかかわらず発言数が少なければ議論をより深めようとしたり、すでに多くの発言が行われているときは論点を絞ろうとしたりするなどの効果も期待できる。

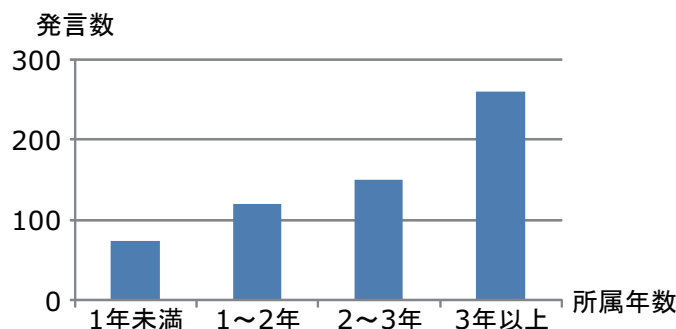


図 4.15: 所属年数による年間発言数の変化

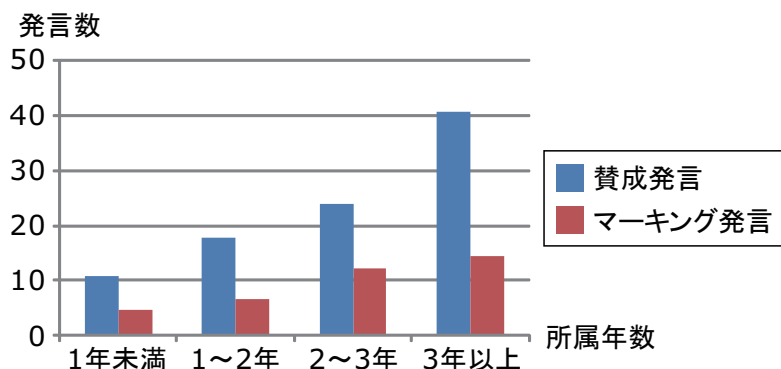


図 4.16: 所属年数による賛成発言およびマーキング発言の数の変化

2番目の項目については、会議の参加者全員が議論に参加し発言を行い易くすることで、結果として議論の活性化につながると考えられる。議論が停滞する原因の一つは、参加者全員が発表に関して同じ知識量を持ち合わせていないことに起因すると考えられる。発表に関する文脈情報や過去の研究情報、あるいは過去に行われた議論内容といった知識を十分に持つ人は、発表内容を容易に理解できるため、議論参加へのモチベーションも高いと予測される。しかし、一方で発表に関して十分な知識を持ち合わせていない人は、説明されない研究の背景やその他の知識に関して差が生じてしまう。これにより発表内容を十分に理解できず、発言を行いにくい状況が発生すると予測される。

上記の内容を確認するため、著者が所属する研究室で2006年度から2008年度までに行われたゼミにおける年間発言数と、ゼミに参加した学生の所属年数との関係を調べた。まず、年度ごとに研究室の学生を所属年数に合わせて、A. 1年未満、B. 1年以上2年未満、C. 2年以上3年未満、D. 3年以上の4グループに分類した。図4.15は、各グループに属する学生の平均年間発言数をグラフにしたものである。たとえば、グループAの研究室に配属されて1年に満たない学生が1年間

で行う発言数は約 75 回であることを表している。所属年数に比例して知識が増していくと考えるならば、このグラフから知識の少ない人ほど発言数が少ないという傾向があることが分かる。図 4.16 は、各グループに属する学生の、自分以外の参加者に賛同された発言（賛成発言）、もしくはマーキングを施された発言（マーキング発言）の数の平均をグラフにしたものである。このグラフからも図 4.15 と同様の傾向があることが分かる。そのため、システムが発表内容を補足する情報を提示することによって、発表に関する知識が少ない人が議論に参加しやすい状況を作り出すことができると考えられる。

以降では、Discussion Recorder によって取得した議論構造の可視化の応用として実現した、現在行われている議論の内容をリアルタイムに可視化するためのツールである Discussion Visualizer と、過去に行われた議論を回顧するためのツールである Discussion Reminder について述べる。

4.2.1 議論中における会議コンテンツの可視化

Discussion Visualizer は、図 4.17 のようなインタフェースであり、サブスクリーンに提示される。Discussion Visualizer は（１）発表情報ビュー（２）スライドリスト（３）議論セグメントビュー（４）議論セグメントリストから構成されている。発表情報ビューでは、発言者を撮影しているカメラのプレビュー、参加者のリスト、発表の経過時間が表示される。

スライドリストは、会議中に表示されたスライドページの一覧が表示される。議論セグメントビューでは、現在行われている発言が含まれる議論セグメントに関する情報が表示される。このビューの上部には、現在行っている議論のきっかけとなる導入発言や、その発言が継続発言であれば親発言のテキストが表示されており、下部には議論セグメントの構造が表示される。議論セグメントリストは、この会議においてどのような議論が行われたかを確認するためのビューである。中央付近には議論セグメント内のサブトピックを表すノードがリスト表示されており、その右側には各サブトピックの内容を表現するキーワードが表示される。ポインタリモコンによって入力された発言に対する賛成の数の多さによってノードの色が変化し、マーキングされた発言を含むノードの右側にはその旨を示すアイコン（図 4.17 内の星のアイコン）が表示され、重要な議論がどの時点で行われたのかを容易に確認できる。スライドリストや議論セグメントリスト、および議論セグメントビューは相互に関連しており、議論セグメントビューで表示されている議論や、その議論に関連するスライドページが議論セグメントリストやスライドリストで強調表示されるようになっている。

3.6.1 項の議論構造の妥当性に関する評価実験の結果から分かるように、議論セグメントの中には、導入発言で提起した話題から派生した内容の継続発言が含まれることがある。つまり、議論セグメントにおいて導入発言をルートノードとして、

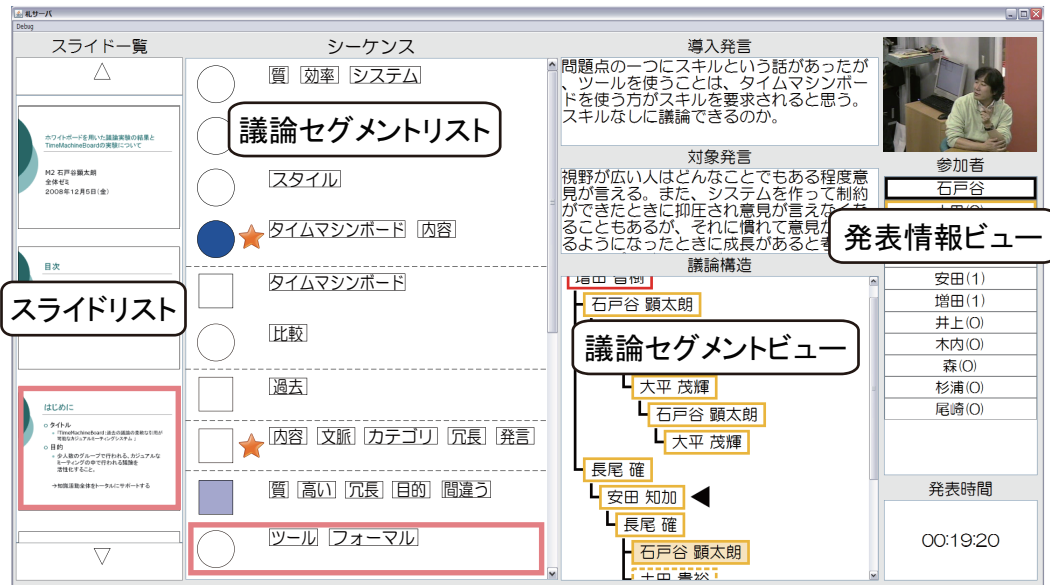


図 4.17: Discussion Visualizer

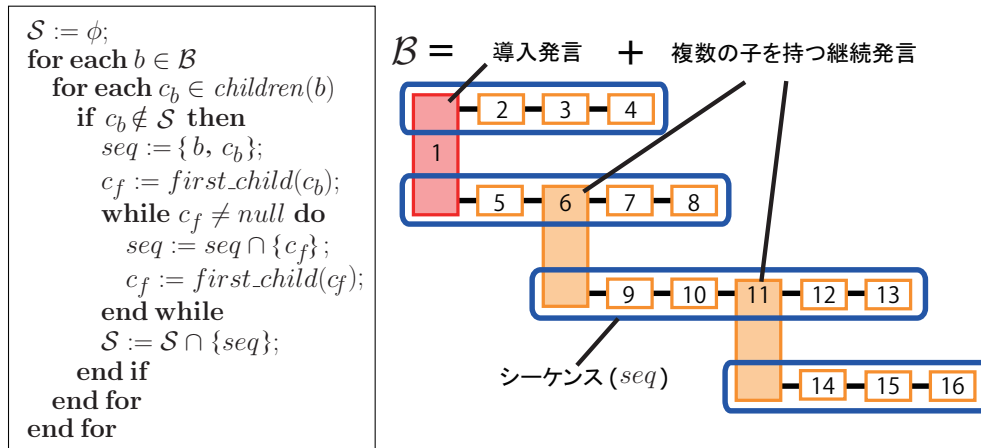


図 4.18: 議論セグメント内のシーケンスとその抽出手順

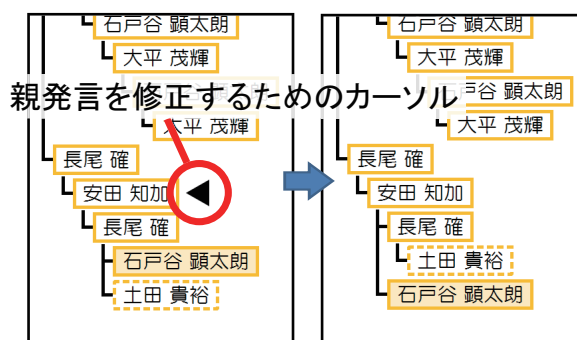


図 4.19: 議論セグメントビューにおける親発言の修正

複数のサブピックが派生していると考えることができる．そのため，Discussion Visualizer では，このような一つの議論セグメントに含まれる複数のサブピックを，図 4.18 に示す手順によってシーケンスと呼ばれる単位の集合 S として抽出している．なお， B は議論セグメント内の導入発言もしくは複数の子発言を持つ継続発言（ブランチと呼ぶ）の集合である．また，関数 $children(s)$ は発言 s に含まれる子発言の集合（子発言が存在しないときは ϕ ）を取得する関数であり，関数 $first_child(s)$ は発言 s に含まれる子発言の中で最も開始時間の早い発言（子発言が存在しないときは $null$ ）を取得する関数である．

3.6.1 項の実験結果では，親発言の異なる継続発言が存在することが分かっている．そこで Discussion Visualizer は，議論構造を可視化するだけでなく，議論構造を編集するための機能を備えている．発言を行っている際に，ポインタリモコンを操作することにより，議論セグメントビューの，議論構造が表示される箇所にカーソルが表示される（図 4.19）．そして，カーソルを正しい親発言に移動させ，選択することによってリンク情報を修正できる．

また，Discussion Visualizer では，ポインタリモコンを用いてスライドリストを操作し，任意のスライドページを選択することで，メインスクリーンに表示されるスライドページを変更できる（図 4.20）．これにより，発言者がスライドのどのページに対して言及しているかを明確に入力できる．

4.2.2 過去の会議コンテンツの回顧

Discussion Reminder を実現するにあたり，我々は以下の 2 点を考慮した．一つは議論内容の正確な把握であり，もう一つは現在の議論を妨げないような高速な議論の検索・閲覧である．曖昧で不十分な情報の共有は，誤解を生み議論の混乱につながる要因になるため，参加者が議論内容を正確に把握できる情報提示が必要である．そのため，Discussion Reminder では，過去の議論の様子を記録したビ

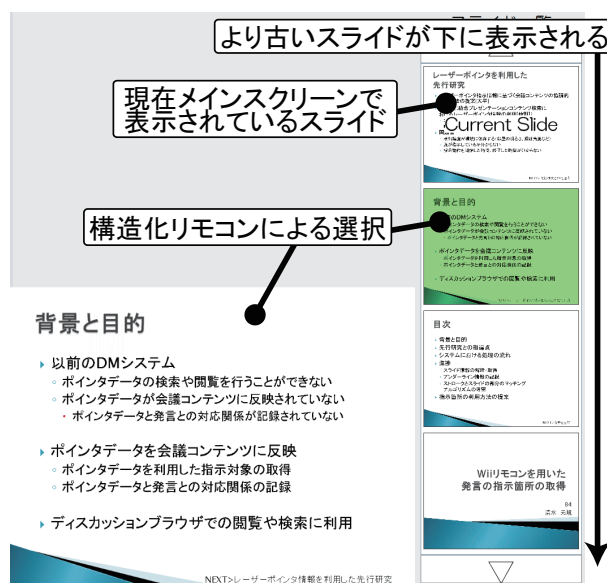


図 4.20: 発言対象となるスライドの選択

デオの閲覧を行う。しかし、参加者全員が過去の議論を検索・閲覧するためには、現在行っている議論を中断する必要があるため、作業にかかる時間は短くしなければならない。そのため、ビデオの中で必要とする箇所をピンポイントに探し出せる手段が必要となる。そこで Discussion Reminder では、クエリにマッチする会議コンテンツ、その会議中に表示されていたスライド、そしてスライドから生まれた発言というように段階的に閲覧したい情報を絞り込むインターフェースを提供する。また複数の参加者が協調的に議論の探索を行う仕組みを提供することによって、より効率的な回顧を可能にしている。

過去の議論の存在を指摘した参加者は書記にクエリを伝え、書記は書記インターフェース上でクエリの入力を行う。クエリには、発表者名、発表日時、キーワードなど Discussion Recorder によって取得されたさまざまなメタデータを指定できる。入力されたクエリに基づいて行われた検索の結果は、図 4.21 のようなインターフェースによってメインスクリーン上に表示され、参加者はポインタリモコンを用いて情報の絞り込みやビデオの再生を行う。

このインターフェースは (1) 会議コンテンツリスト (2) スライドリスト、(3) 議論セグメントリスト (4) 議論セグメントビューの 4 つで構成されている。会議コンテンツリストでは、クエリにマッチした議論を含む会議コンテンツのタイトル一覧が表示される。ポインタリモコンによってタイトルを選択すると、選択された会議コンテンツに含まれるスライドページのサムネイル一覧がスライドリストに表示される。スライドリストでは、サイズの大きいサムネイルのプレビューを行うこともできる。そして、Discussion Visualizer と同様に、スライドリストや議

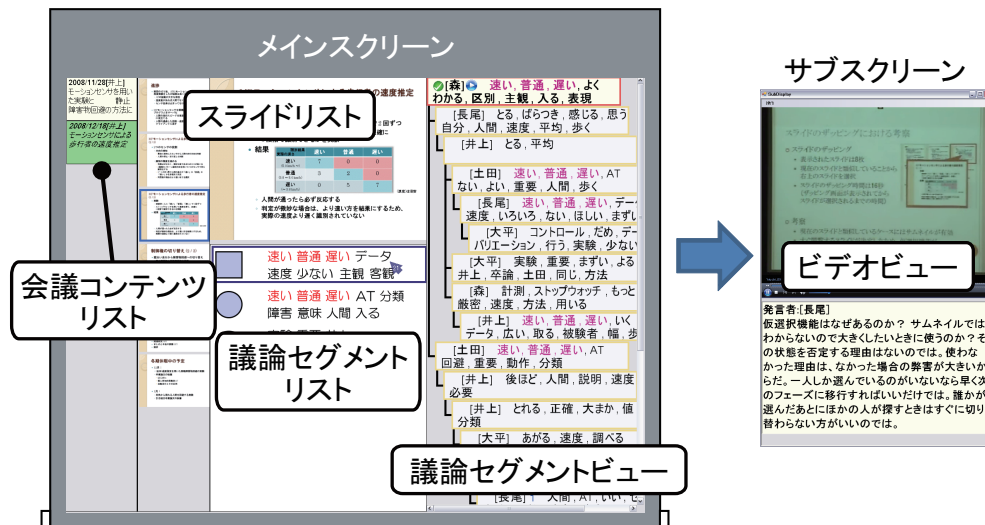


図 4.21: Discussion Reminder

論セグメントリスト、議論セグメントビューは相互に関連しており、左側のコンポーネントで選択されたオブジェクトに同期して、右側のコンポーネントの表示が動的に変化する。そして参加者は、これらの情報を確認しながら、ポインタリモコンによって閲覧したい発言を議論セグメントビュー上で選択する。発言を選択することで、サブスクリーン上に表示されているビデオビューで対応する時間からビデオを閲覧できる。

Discussion Reminder では、複数の参加者が協調的に議論の探索を行うことができる。具体的には、ビデオビューで再生をしている最中でも発言テキストやスライドページのプレビューを行うことができる。ある参加者が間違った発言を再生したとしても、別の参加者がプレビューを行うことによって正しい発言の存在を指摘できる。また、必要に応じて対象とするスライドページを変更することもできる。

しかし、複数の参加者が競合する操作を同時に行うことによって混乱が生じる可能性がある。たとえば、ビデオビューで再生を行っている最中に、別の参加者が別の発言の再生を開始すると、それまでの再生は中断されてしまう。そのため、Discussion Reminder ではビデオビュー操作をロックする機能が用意されている。ビデオビューで再生を始めると、再生を開始した参加者が停止させるか、その発言が終了するかまで他の参加者は発言テキストやスライドページサムネイルのプレビュー以外の操作を行うことができなくなる。これにより、複数の参加者が競合することなく、効率的に発言内容を確認できる。

そして、過去の発言を回顧しながら新たな議論セグメントが作成された場合は、参照した発言と作成された議論セグメント内の導入発言間のリンク情報が生成・記

録される．これにより，複数の会議コンテンツにまたがる要約などの応用を実現できる．

4.2.3 ユーザビリティに関する考察

ここまでは，議論を円滑するためのアプリケーションである Discussion Visualizer および Discussion Reminder について述べた．しかし，現段階で長期間にわたる運用が行えておらず，これらのアプリケーションの定量的な評価までには至っていない．そこで本項では，それぞれのアプリケーションの運用を通じて得られた，ユーザビリティに関する考察について述べる．

Discussion Visualizer

Discussion Visualizer では，発言タイプや発言予約機能によって生成される議論構造を可視化する機能を持っている．この議論構造の可視化を行うことで分かったことは，参加者が積極的に議論構造の誤りを修正しようとすることである．これまでは，自分の発言が他のどの発言に対して行っているものなのか分からずに議論していたが，Discussion Visualizer によって，議論構造の誤りを直接確認できる上，ポインタリモコンのボタンを用いた簡単な操作でその誤りを修正できるため，発言者が積極的にリンク情報を修正する場面が多々見られた．

また，ポインタリモコンを用いて入力された発言に対する賛成やマーキング情報を可視化することで，発言そのものの評価が積極的に行われる場面も確認された．発言に対する評価をリアルタイムで確認できるため，発言者は発言に対する達成感を得ることができる．また，賛成情報が入力されると発表情報ビュー内の参加者名が点滅するようになっているため，賛成の度合いが強い場合に連続してボタンを押下すると，参加者名が激しく点滅する．その視覚効果が他の参加者にとっても楽しい印象を与えているという現象も確認されている．

このように，Discussion Recorder によって取得している情報をリアルタイムに表示することによって，議論そのものに積極的に参加できるようになると考えられる．

Discussion Reminder

Discussion Reminder を用いて過去の会議コンテンツを回顧することで顕在化したのは，対象となる議論の存在を把握している参加者と把握していない参加者との間に存在する知識レベルの差異だった．実際，回顧の実施を促したり操作を先導したりしていたのは，知識レベルが高く議論の存在を把握している参加者だった．このような参加者は背景知識があるため，Discussion Reminder で提示される

議論内容を閲覧するだけで大まかな内容を想起することができる．これに対して知識レベルの低い参加者は，発言テキストやビデオを閲覧するだけではその内容を深く理解することができない．それは，試験的に出題した過去の議論に関する穴埋め形式の設問に対して，知識レベルの低い参加者は表層的な回答はできても，その議論に関する文脈情報までは理解できなかったという結果からも推測できる．

以上のことから，現状の Discussion Reminder を用いた会議コンテンツの回顧だけでは，会議中に全ての参加者を同等の知識レベルにすることは困難であると考えられる．しかし，Discussion Reminder では参照した議論とその後に行われた議論を関連付けることができるため，会議後にその関連情報と合わせて Discussion Browser を用いて詳細に内容を確認することで，知識レベルの差異を埋めることはできると考えられる．

4.3 本章のまとめ

本章では，議論フェーズにおいて DRIP システムが提供するいくつかのアプリケーションについて述べた．具体的には，会議コンテンツの検索・閲覧を行うための Discussion Browser，議論中に会議コンテンツの可視化や回顧を行うことによって，議論の円滑化・活発化を支援するための Discussion Visualizer や Discussion Reminder を提案した．

Discussion Browser は，Discussion Recorder によって取得したメタデータの俯瞰や，内部構造の検索を行うことで，効率的に会議コンテンツの閲覧を行うことができるアプリケーションである．しかし，会議コンテンツの利用状況に関するアンケート調査を行うことによって，会議コンテンツの作成・検索・閲覧だけでは，知識活動を十分にサポートできていないことが明らかになった．

また，取得したメタデータをリアルタイムで可視化する Discussion Visualizer や，過去の会議コンテンツの回顧を行うための Discussion Reminder についても述べた．これらのアプリケーションの運用に基づく考察から，メタデータの可視化によって参加者がより積極的に議論に参加できること，会議コンテンツの回顧による知識の共有化がいかに困難であるかが分かった．

次章では，Discussion Recorder によって作成された会議コンテンツを再利用する仕組みについて述べる．

第5章 会議コンテンツの再利用による知識活動支援

3章, 4章では, 議論フェーズで行われた議論の内容を会議コンテンツとして記録する Discussion Recorder, 会議中の議論を円滑するために会議コンテンツの可視化・回顧を行う Discussion Visualizer や Discussion Reminder, および作成された会議コンテンツを検索・閲覧するための Discussion Browser について述べた。しかし, 知識活動を円滑に行うためには, 会議コンテンツの作成・検索・閲覧だけではなく, それを再利用できる仕組みが必要となる。また, 会議コンテンツの再利用を通じて創造されたアイデアや知識を記録し, 会議コンテンツと関連付けることで, 計算機が個人の知識活動に関する情報を把握・利用できるようになる。

本章では, 議論フェーズ後の3つのフェーズ(再認フェーズ, 探求フェーズ, 集約フェーズ)において, DRIP システムが提供する機能について述べる。これらの機能は全て, 2章での図 2.10 における Discussion Mediator 上で実現されている。まず, DRIP システムが記録・蓄積しているデータについて述べたのち, 議論後の各フェーズにおける支援について述べる。最後に, DRIP システムの有効性を確認するために行った評価実験の結果とそれに対する考察についても述べる。

5.1 DRIP システムにおける知識活動マップの作成と その利用

5.1.1 知識活動マップの作成

DRIP システムでは, さまざまな種類のコンテンツの作成, 編集, 蓄積を行うことで, 知識活動マップの作成を行っている。DRIP システムで扱うコンテンツを以下に示す。発表者がこれまでに創造してきたアイデアや知識に関する報告を行うために用いられる発表資料も, 同様の観点からコンテンツとして DRIP システム上で作成, 編集を行う。

- 会議コンテンツ
- 発表資料

- ノート
- TODO

しかし，本研究では，会議コンテンツや発表資料のように多数の人間に対して公開されているものだけではなく，メモ書きのように個人のためだけに作成されたものもコンテンツとして扱い，記録を行う．このようなコンテンツは，最初に作成された時点では，個人の覚え書き程度の内容でしかなく，作成者以外の人間が見てもその内容が理解できないようなものであるかもしれない．しかし，その後のさまざまなタスクを通じて，その内容は随時追加・更新され，発表資料や論文といった，他人に公開するためのコンテンツの材料として利用される．また，企業におけるプロジェクトのように複数の人間がグループとして，共通のテーマに関する知識活動を行っている場合は，メンバー間で意見交換やアイディアの共有を行うことによって，グループ全体の活動が円滑に行うことができ，生産性を高めることができると考えられる．これらのことから，本研究では，新たに生まれたアイディアや知識をノートと呼ばれるコンテンツとして記録・蓄積する．

また，会議コンテンツ内の議論内容を踏まえて，次の会議までにやらなければならないタスクを決定したり，さまざまなタスクを通じて新たなアイディアや知識を創造したりするように，知識活動においてタスクは重要な意味を持っている．そこで，DRIP システムでは，このようなタスクもコンテンツの一種として捉え，TODO と呼ばれるデータとして記録・蓄積を行う．

再認フェーズで議論内容の反芻を行ったり，探求フェーズで新たなアイディアや知識を生み出したりする際，コンテンツを分類することで効率的に必要な情報を参照できる．コンテンツの分類を行う主体として，DRIP システムとその利用者の2種類がある．前者による分類手法として自然言語処理技術があるが，知識活動を通じて生み出される新たなアイディアや概念に対応するためには辞書の永続的な更新が必要であり，DRIP システムを運用するにあたって障害となる可能性がある．また，過去の議論内容を参照する際，利用者にとって重要度が有効な場合があるが，現在の自動認識技術では，人間の解釈を踏まえた上で意味内容を解釈することは困難である．そこで，自動認識技術では取得が困難な情報は，利用者が入力する方法が必要となる．それを解決する手段の一つとして，アノテーションと呼ばれる技術がある [12]．アノテーションとは，図 5.1 のように文書や画像・ビデオ等のコンテンツに付与されたメタデータを意味する．

DRIP システムで取得しているアノテーションには，以下のものがある．

- キーワードタグ
- マーキング（会議コンテンツに対してのみ）

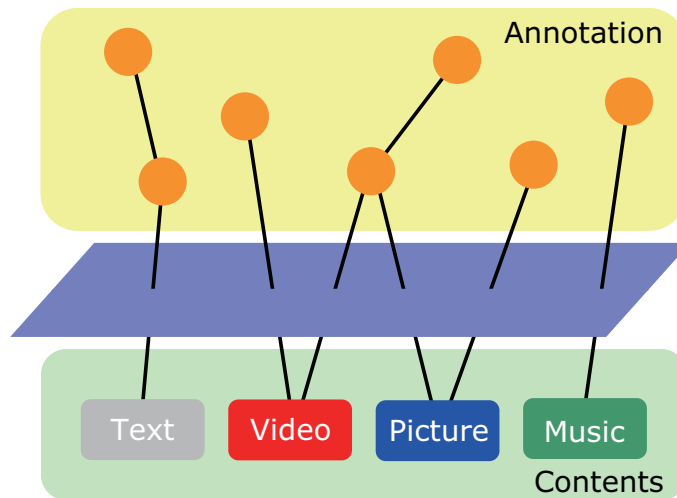


図 5.1: アノテーションの概念図

キーワードタグ キーワードタグは、はてなブックマーク¹や Delicious²に代表されるソーシャルブックマークのタグのように、コンテンツの内容を端的に表現するキーワードである。これにより類似するコンテンツ同士をまとめることができ、新たなアイディアや知識を創造する際に必要となるコンテンツの検索が容易になる。

時間の経過と共に議論の内容が忘れ去られてしまうように、自身の活動内容も忘れ去られてしまうことが考えられる。そのため、過去のコンテンツが忘れ去られ、同じ内容のコンテンツが新たに作成されてしまう可能性がある。そのため、キーワードタグを付与することによって、蓄積されているコンテンツを整理し、冗長なコンテンツの作成を防ぐことができる。

マーキング（会議コンテンツに対してのみ） 知識活動において議論は繰り返し行われるものであり、作成される会議コンテンツは次第に増えていく。しかし、作成された会議コンテンツ内のすべての発言は同等の重みで扱われるため、自分にとって重要な発言がどの会議コンテンツに含まれているのかが分からなくなる可能性がある。つまり、自分が重要であると判断した発言に対して何らかの情報を付与することが望ましい。それを実現するのがマーキングである。

ある発言が自分にとって重要であるという判断は、議論中に行うものと議論後に行うものがあると考えられる。議論中に判断した発言は、Discussion Recorder で利用しているポインタリモコンによって付与できる。また、議論後にマーキングを施すには Discussion Browser を用いる（詳細は後述）。

¹株式会社はてな：はてなブックマーク．<http://b.hatena.ne.jp/>（2010 年 12 月 11 日）

²Yahoo! Inc.：Delicious．<http://del.icio.us/>（2010 年 12 月 11 日）

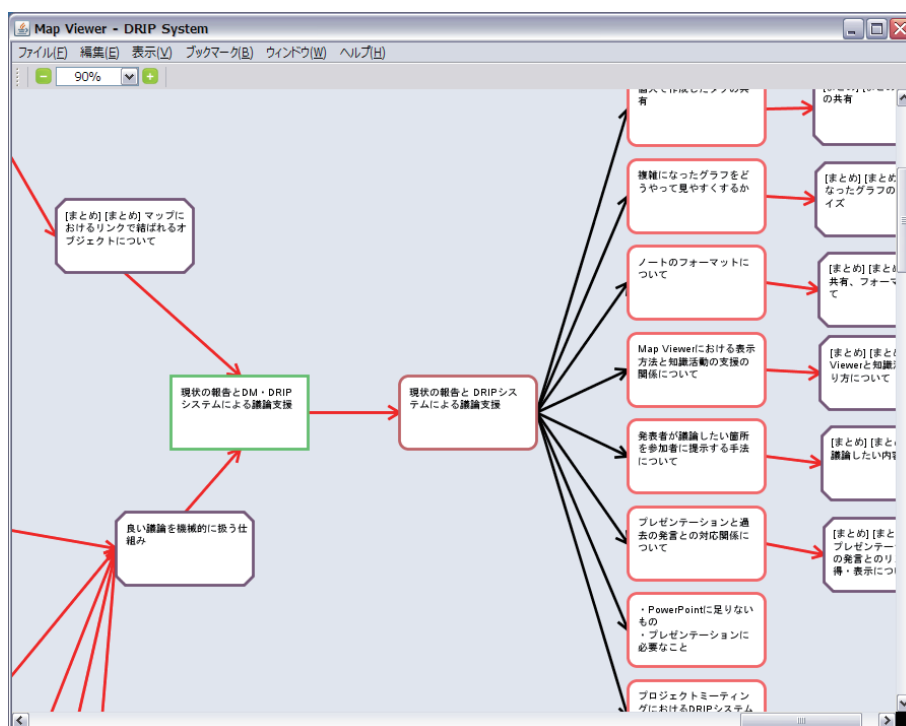


図 5.2: 記録・蓄積された知識活動マップの可視化

5.1.2 知識活動マップのもたらす効果

DRIP システムによって記録・蓄積された知識活動マップは、図 5.2 のような表示で確認できる。また利用者は、可視化されたノードの配置を変更することで、知識活動マップそのものの編集を行うことができる。このようなインタフェースを提供することで、以下に示す効果が期待される。

知識活動の俯瞰および円滑化 テキストや図を用いて、思考状態をグラフィカルに表現する手法としてマインドマップ³がある。これは、中心となるトピックから線を引き、その線の上にそのトピックから派生するテキストや図を書き足すことで非線形・放射状のマップを作成し、人間の発想を支援する手法である。人間の頭の中で行われている思考に近い内容を目に見える形式で出力することによって、情報の記憶・整理・理解・発想をより豊かに行うことができる。マインドマップにおいてトピックに関するテキストや図を派生関係に基づいて可視化するように、知識活動マップ内のコンテンツ（ノード）を、引用によって作成されるコンテンツ間の派生関係（リンク情報）に基づいて可視化することで、自身の知識活動の

³株式会社 ALMACREATIONS：マインドマップ。
<http://www.mindmap.ne.jp/>（2010 年 12 月 11 日）

俯瞰や整理が行いやすくなる。たとえば、ノードを時系列順に配置することで活動の流れを分かりやすくしたり、同じタスクに関連するノード同士を近い位置に配置することで視覚的な分類を行ったりできる。また、ノードの配置の上下関係を利用して、タスクの優先度を直感的に決定することもできるだろう。

それだけでなく、計算機が知識活動マップを分析することによって、利用者の知識活動の円滑化を支援できる。たとえば、参照情報が付与されていない会議コンテンツが存在する場合、それは存在を忘れられ放置されている会議コンテンツだと捉え、利用者にその存在を提示することによって、議論内容の再確認・整理を喚起できる。また、コンテンツの最終更新日時から一定以上の期間更新されていないコンテンツの存在を提示することで、タスクの放置を減らすことができる。

関連コンテンツの発見支援 知識活動マップ内で保持されているコンテンツ間の参照情報やキーワードタグを利用することで、タスクの内容に関連するコンテンツの発見を支援できる。たとえば、過去の会議中に出たシステムに対する要望を実装する際、過去の議論内容を参照することがある。従来では、Discussion Browser上で、参照したい議論が含まれる会議コンテンツをキーワード検索によって探し、さらに会議コンテンツ内の発言内容を確認するという作業が必要だった。これに対して、DRIP システムでは、要望を実装するという TODO と、そのきっかけとなる発言集合との間に参照関係が記録されているため、その情報を辿ることで効率的に必要とする内容を確認できる。

また、コンテンツに対して付与されているキーワードタグを利用することで、参照情報だけでは到達することができないような関連コンテンツを取得できる。たとえば「会議コンテンツ」「アノテーション」というキーワードタグが付与されているコンテンツ群を取得すると、そこには数か月、数年も以前に行われた同様の内容の会議コンテンツが含まれている可能性がある。その会議コンテンツの内容を再び確認・引用することによって、新たな参照情報が追加される。

次節では、議論フェーズ後の3つのフェーズにおいて、DRIP システムが提供する機能について述べる。

5.2 議論内容の整理とタスクの決定

再認フェーズでは、Discussion Recorder によって作成された会議コンテンツの内容を整理する作業である、リフレクションを行う。リフレクションを行うための流れを以下に示す。また、リフレクションを行うことによって、作成される知識活動マップの例を図 5.3 に示す。

1. 作成された会議コンテンツの取り込み

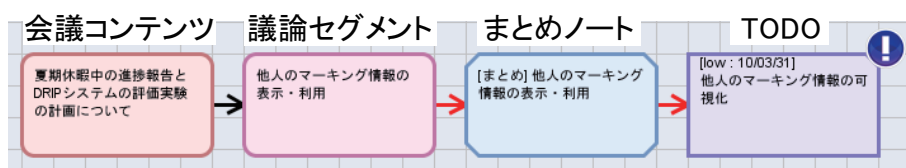


図 5.3: リフレクションによって生成された知識活動マップ

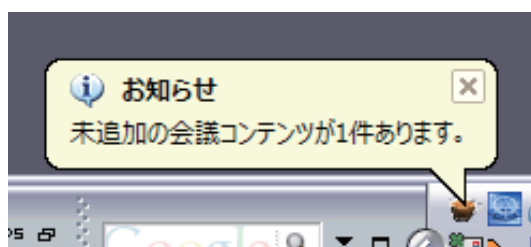


図 5.4: 会議コンテンツの追加を知らせるメッセージ

2. Discussion Browser 上でのマーキングの付与
3. 議論セグメントの取り込み
4. 議論のまとめの記述・キーワードタグの付与
5. TODO の作成

5.2.1 作成された会議コンテンツの取り込み

新しい会議コンテンツが作成された後、Discussion Mediator を起動すると、Discussion Mediator は Web サーバに新規に追加された会議コンテンツがないか問い合わせを行う。もし、新規に追加された会議コンテンツが存在した場合、Discussion Mediator は、図 5.4 のようなメッセージを表示する。メッセージを表示することによって、リフレクションのし忘れを防ぐことができる。このメッセージを選択することで、追加された会議コンテンツを Discussion Mediator 内への取り込みが行われる。

5.2.2 マーキングの付与

取り込まれた会議コンテンツを選択すると、Discussion Browser が起動し、その会議コンテンツの内容を閲覧できる。Discussion Browser の議事録ビュー（図 4.7

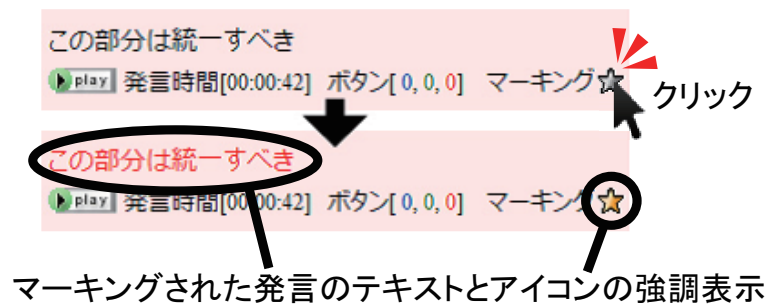


図 5.5: Discussion Browser 上でのマーキング

参照)では、発言ごとにマーキング情報を表すアイコンが表示されている(すでに会議中に付与されたマーキングは強調される)。マーキングが施されていない発言のアイコンをクリックすることで、会議中には施すことができなかったマーキングを、会議後に施すことができる(図 5.5)。

5.2.3 議論セグメントの取り込み

Discussion Recorder では、議論を話題単位に分割した単位である議論セグメントを作成している。そこで、DRIP システムでは、同一の議論セグメント内に含まれる、マーキングが施された発言をまとめて取り込むことによって、複数発言の引用を容易に行うことができる。議事録ビュー上で、取り込みたい議論セグメント(に含まれる発言)をメニューから選択すると、Discussion Mediator が選択された議論セグメントに関する情報を Web サーバから取得し、ノートの記述やキーワードタグの編集を行うためのインタフェースが表示される(図 5.6)。このインタフェースは、選択された議論セグメントとそれに関連付けられたスライドサムネイルやマーキングが施された発言のリスト(左側)、キーワードタグの編集を行うためのコンポーネント(中央)、そしてまとめを記述するコンポーネント(右側)から構成される。

会議コンテンツの中には、同一の話題に関する議論が行われているにも関わらず、途中で別の話題に関する議論が行われた(新たな導入発言によって別の議論セグメントが生成された)ために、別々に分割された議論セグメントが存在することがある。そこで、一方の議論セグメントを取り込み、図 5.6 のインタフェースを表示している状態で、別の議論セグメントを取り込むことで、同一の内容の議論セグメント同士を結合できる。

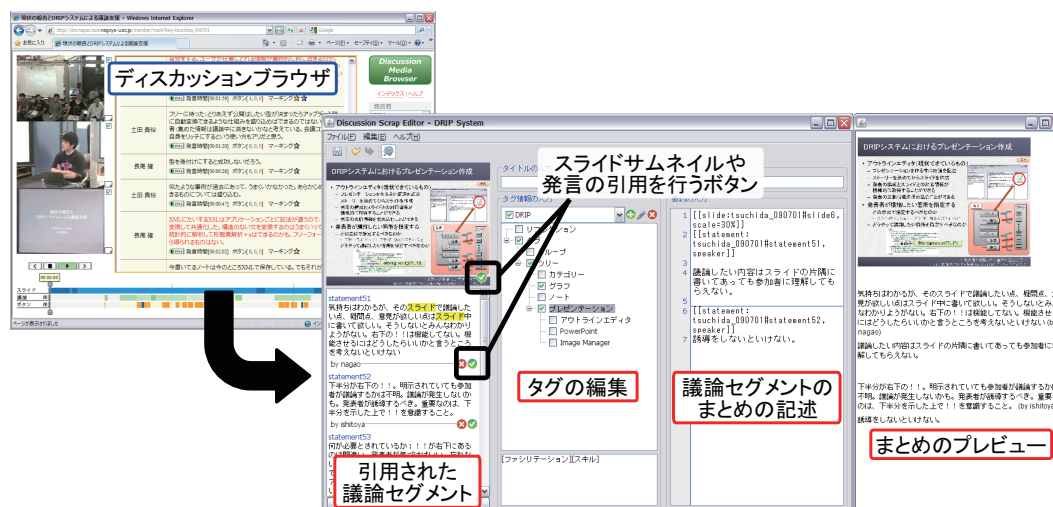


図 5.6: Discussion Browser と Discussion Mediator との連携

5.2.4 議論のまとめの記述・キーワードタグの付与

図 5.6 のインタフェースの右側には、選択した議論セグメントのまとめを記述するコンポーネントがあり、ここで記述されたまとめはノートして保存される。このコンポーネントでは、リスト構造や表を利用したノートの構造化や画像の表示を行うことができ、イメージの表現や物事の対比をしやすいとしている。それ以外にも、ハイパーリンクの自動作成やフォントの大きさや色の変更による文字列の装飾などを行うことができる。

まとめには、書記の入力した発言テキストだけでは不足している議論内容の補足や、自分なりに解釈した発言内容などを記述する。その際、「このスライドの表現を受けてこのような発言が行われたのではないか」「この発言でこの人はこういうことを言っていたのではないか」といったように、スライドサムネイルや発言情報そのものを引用しながらまとめを記述できることが望ましい。そのため、インタフェース左側のスライドサムネイルや発言のリストには、引用を行うためのボタンが表示されており、必要に応じて引用を行うことができる。実際にスライドサムネイルや発言情報を引用して作成されたまとめの例を図 5.7 に示す。

取り込んだ議論セグメントやまとめを分類するためのキーワードタグは、ユーザが自由に入力できるだけでなく、発言テキストから必要な箇所を選択して入力することもできる。

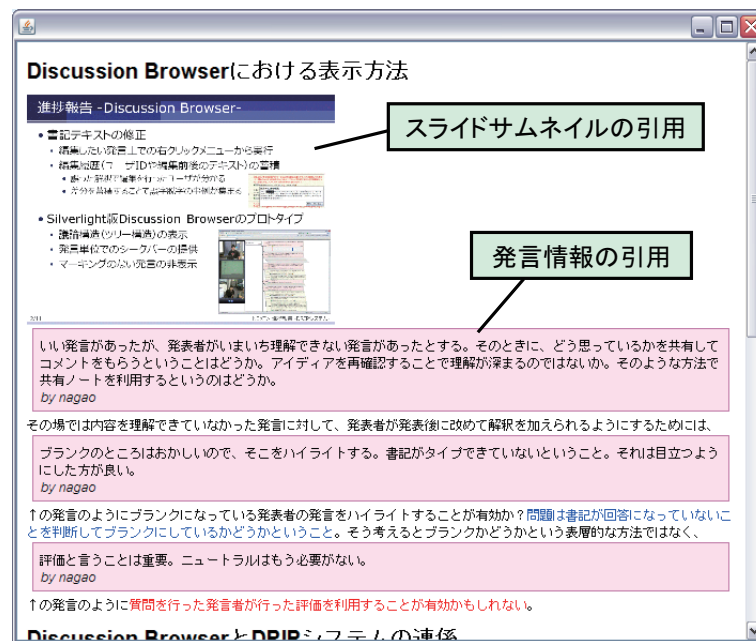


図 5.7: スライドサムネイルや発言情報を引用したまとめノート

5.2.5 TODO の作成

議論内容をまとめることによって、自身がこれからどんなタスクを遂行するべきなのかが明確になった場合、利用者はまとめと関連付けながら TODO を作成できる。図 5.6 の議論セグメントのまとめを記述するコンポーネント内で、特殊な記法を用いて TODO へのリンクを記述した状態で保存を行うと、図 5.8 のようなインタフェースが表示される。利用者は、TODO のタイトルや締め切り、重要度などの情報を入力することで TODO を作成する。

5.3 整理した議論内容に基づくタスクの遂行

5.3.1 ノートの作成・編集・蓄積

探求フェーズは過去の議論内容を踏まえつつ、調査や実装、検証といったさまざまなタスクを行うことで新たな知識を得たり、新たなアイデアを創出したりしていく。DRIP システムでは、このようなタスクを通じて得られた知識やアイデアをノートとして記録・蓄積を行う。なぜなら、知識活動の成果をノートとして記録することによって、自身の知識活動がどれほど進展しているかを確認できるだけでなく、このようなノートを引用しながらスライドを作成することで、スライドの作成コストを下げることもできるからである。

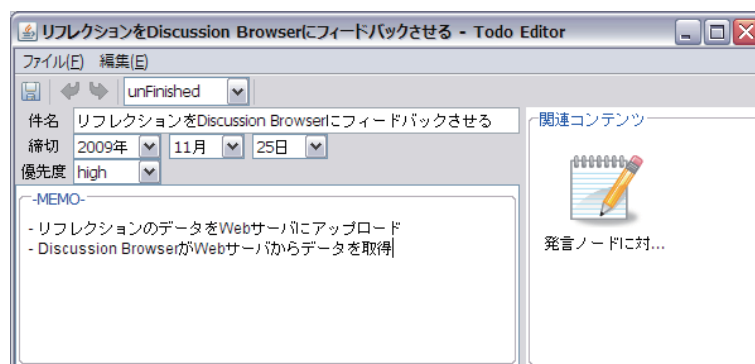


図 5.8: TODO の作成インタフェース

図 5.3 の TODO ノードを右クリックして表示されるメニューからノートを作成を行うと図 5.9 のようなインタフェースが表示される。再認フェーズと同様に、ハイパーリンクの自動作成やリスト構造・表の利用など構造化されたノートを記述できる。そして、作成されたノートはタスクと関連付けられ、他のコンテンツに引用可能な状態となる。

5.3.2 コンテンツ間の参照情報の取得

図 5.9 のインタフェースは、議論セグメント内の発言テキストや関連するノートのテキストのコピーアンドペーストによって、自動的にコンテンツ間のリンク情報を生成し、記録する機能を持っている。たとえば、参照元となるノートのテキストの一部を選択・コピーすると、クリップボード上に選択されたテキストだけでなく、そのノートの識別子も格納される。そして、参照先となる別のノートの編集インタフェース上でペーストしたとき、編集インタフェースはクリップボード内のデータに選択されたテキスト、および参照元のノートの識別子があることを認識することで、テキストの貼り付けだけでなく参照関係を自動的に生成・記録できる。追加された参照情報は、図 5.9 の中央部に表示され、必要に応じてその内容を確認できる。

この参照情報を生成する手法は、ノートだけに限らず知識活動マップで扱うコンテンツ全てに適用できる。なぜなら、クリップボード内に参照元の識別子というコンテンツに共通する情報だけを格納することで、参照情報を生成できるからである。もちろん、クリップボード内にはテキスト情報も格納できるため、通常のテキストエディタにもペーストできる。そのため、既存のツールと DRIP システムは競合することなく、データのやり取りを行うことができる。

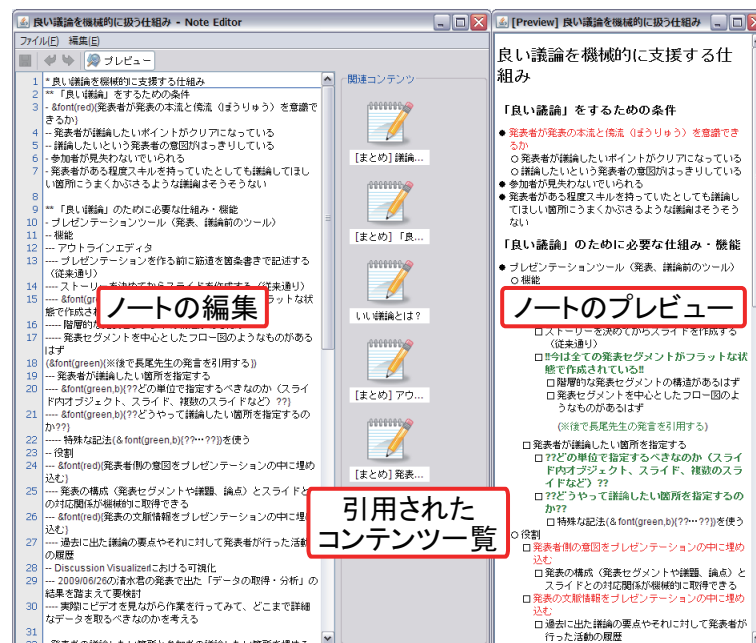


図 5.9: ノート編集インタフェース

5.3.3 ノートと TODO との関連付け

文献調査というタスクを通じて分かった問題点を解決するために、解決策のアイデア出しやシステムの実装を行うように、タスクを通じて得られた知見から新たなタスクが生まれることがある。利用者は、タスクを遂行する中で作成したノートと関連付けながら、新たな TODO を作成できる。

5.4 蓄積した情報を用いた発表資料の作成支援

ゼミのように発表・議論が繰り返し行われる場合、過去の議論内容を受けて、どのような活動を行い、どのような成果を得られたのかを説明することによって、参加者の発表内容に対する理解が容易になり、結果としてより活発な議論を行うことができると考えられる。そのため、前述の議論フェーズで作成された会議コンテンツや探求フェーズで作成されたノートを引用しながらスライドを作成することは有効だと考えた。そのため DRIP システムでは、集約フェーズにおけるスライドの作成を支援するため、以下に示す機能を持つインタフェースを提供している(図 5.10)。

- アウトラインエディタ (発表構成の編集)
- 過去のコンテンツの引用

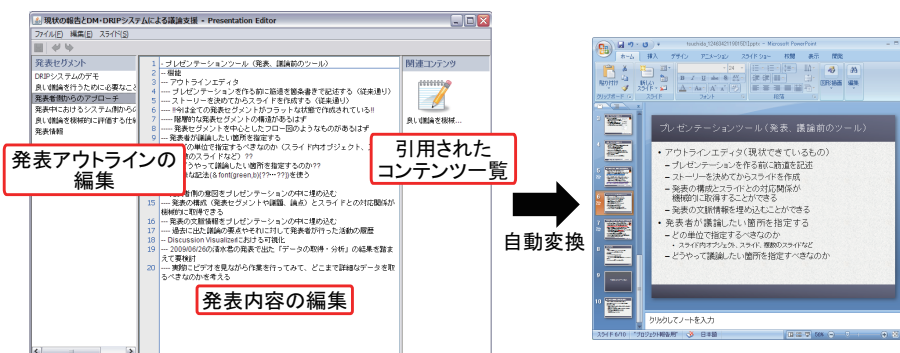


図 5.10: スライド作成インタフェース

● スライドファイルへの自動変換

自身の活動内容を効率的に伝達する上で、「背景・目的 アプローチ 実験結果 考察 まとめ 今後の課題」のように発表構成を意識しながらスライドを作成することは有効である。そのため、図 5.10 の左側には発表構成を編集するためのコンポーネントが用意されており、発表の大まかな内容を作成したり順序を入れ替えたりできる。本インタフェースの中央部は、発表内容を詳細に記述するためのエディタになっている。ユーザは、ノートの時と同様に発表内容を構造的に記述することができ、過去のコンテンツのテキストをコピーアンドペーストすることで自動的にコンテンツ間のリンク情報を記録できる（記録された情報は図 5.2 のインタフェース上で俯瞰できる）。

そして、本インタフェースは、作成された発表内容を Microsoft PowerPoint 形式のスライドファイルへ自動変換できる。自動変換を行う際、構造情報を持った発表内容とコンテンツの引用情報が記録された XML ベースの中間言語を生成する。スライドファイルはデザインテンプレートやアニメーションのように発表内容と関係のない情報を含んでいるが、生成される中間言語は構造情報を持った発表内容とコンテンツの引用情報だけを含んでいるため、スライドファイル以外のコンテンツへの再利用が可能である。たとえば、構造情報を LaTeX⁴の書式で変換することで論文に、Microsoft Word 形式のファイルに変換することで報告書に再利用できる。そして、自動変換されたスライドは、プレゼンテーションサーバへのアップロード、および Discussion Recorder を用いた発表への利用が可能である。アップロードされたスライドを用いて行われた発表内容に対して議論が行われることで、新たに作成された会議コンテンツ内の議論セグメントとスライドが関連付けられる。

⁴LaTeX . <http://www.latex-project.org/> (2010 年 12 月 11 日)

5.5 評価実験

提案した DRIP システムによって、会議コンテンツとゼミ後のタスクやタスクの成果物とを関連付け、蓄積されたさまざまなコンテンツの情報をを用いて発表資料を作成できる。本節では、DRIP システムによってより有意義なゼミを行うことができたかどうかを確認するために行った評価実験について述べる。

5.5.1 実験環境

DRIP システムの有効性を確認するため、DRIP システムを用いずに作成された発表資料を用いて行われたゼミと DRIP システムを用いて作成された発表資料を用いて行われたゼミとの比較実験を行った。被験者は著者が所属する研究室の大学生・大学院生から発表者として 2 名、参加者として 3 名の計 5 名を選出した。発表の経験に差が出ないようにするため、同時期に研究室に配属された 2 名を発表者として選出した。発表者を DRIP システム利用者（以下、発表者 A）と非利用者（以下、発表者 B）に分け、それぞれの発表者は、自身の研究活動に関する発表を 60 分を目安に計 5 回行った（発表は 4 日から 6 日の間隔をおいて実施した）。全ての発表は Discussion Recorder を用いて行われ、発表者の利用できる発表資料は Microsoft PowerPoint のスライドファイルのみとした。また、書記による発言テキストの量や質の個人差を最小限にするため、すべてのゼミにおいて同じ被験者が書記を行うようにした。

それぞれの発表者は、ゼミ中はポインタリモコンを用いて、ゼミ後は Discussion Browser を用いてマーキングを施してもらった。また、発表者 A にはマーキングを施した後、DRIP システムを用いて会議コンテンツに対するタグの付与や、ノート作成、発表資料の作成を行い、発表者 B は DRIP システムを用いずに発表資料の作成を行った。また、ゼミが行われた直後にゼミの参加者に対してアンケートを行い、前回行われた議論が発表資料や発表内容に反映されているかどうかを確認してもらった。

また、発表内容に対する影響を最小限にするために、以下のことを考慮して実験を行った。

- 発表者と参加者はゼミ以外の時間に発表内容に関する議論を行わないようにする。
- 発表資料が作成される前に記入されたアンケートを回収し、発表者にはアンケート結果を公開しないようにする。

5.5.2 評価手法

本実験では、過去の会議コンテンツをふまえた発表ができたかどうかを評価するため、直前のゼミで行われた発言の内容が次の発表に十分に反映されているかどうかを調べる。ここでは、直前のゼミで行った参加者の発言を、その内容と次の発表内容を照らし合わせながら、以下に示す5つのカテゴリに分類する。

- (1) 次の発表に反映してほしかった内容であり、それが自分の意図どおりに反映されていた。
- (2) 次の発表に反映してほしかった内容であり、スライドや発表内容で触れられてはいたが、自分の意図した内容に沿うものではなかった。
- (3) 次の発表に反映してほしかった内容であるにもかかわらず、スライドや発表内容で触れられていなかった。
- (4) 今すぐ反映してほしいわけではなく、今後反映してほしい内容である。
- (5) 単純な質問などのように上記のカテゴリにあてはまらない内容である。

本実験では、DRIPシステムの有無によって上記のカテゴリ、特に(1)や(2)、(3)に該当する発言の数が異なるかを調べる。なお、(1)や(2)、(3)のいずれかに該当する発言は、発表者が次の発表に反映してほしかった内容を含むものであるため、以下ではこれらをまとめて要対応発言と呼ぶ。(1)に該当する発言数が多い発表ほど有意義な発表ができたと考え、逆に(2)や(3)に該当する発言数が多い発表ほど有意義な発表ができなかったと考える。

以下ではこの分類を行うための手順について述べる。はじめに、発言者の「次の発表に反映してほしい」という意図を取得するため、1回目から4回目までのゼミが終了した直後、各参加者に直前のゼミで自身が行った発言の一覧を配布した。そして各参加者に、発言内容を思い出しながらそれぞれの発言を以下の3つに分類してもらった。

- a. 次の発表に反映してほしい内容である（要対応発言）。
- b. 今すぐ反映してほしいわけではなく、今後反映してほしい内容である（カテゴリ(4)）。
- c. 単純な質問のように上記のカテゴリにあてはまらない内容である（カテゴリ(5)）。

そして、2回目以降のゼミが終了した直後、前述のアンケートとは別に、各参加者に前回のゼミで自身が行った発言の一覧と今回のゼミで使用された発表資料のハンドアウトを配布した。そして参加者に、発表内容を思い出しながら前回のゼミで自身が行った要対応発言を以下に示す3つに分類してもらった。

表 5.1: 実験で行われたゼミの時間（分）

発表者	実施回					合計	平均
	1	2	3	4	5		
A（利用者）	51.8	60.0	70.9	70.1	52.9	305.7	61.1
B（非利用者）	60.4	61.4	57.9	68.2	65.0	312.9	62.6

表 5.2: 会議に行われた発言数および議論セグメント数（括弧内は議論セグメント数）

発表者	実施回					合計	平均
	1	2	3	4	5		
A（利用者）	27(5)	29(6)	30(8)	40(10)	34(10)	160(39)	32.0(7.8)
B（非利用者）	30(9)	23(8)	26(6)	34(14)	32(5)	145(42)	29.0(8.4)

- i. 発言内容が自分の意図どおりに反映されていた（カテゴリ (1)）。
- ii. 発言内容についてスライドや発表内容で触れられてはいたが，自分の意図した内容に沿うものではなかった（カテゴリ (2)）。
- iii. 発言内容がスライドや発表内容で触れられていなかった（カテゴリ (3)）。

最後に，それぞれのアンケート結果を集計することで，1 回目から 4 回目までの発言を前述の 5 つのカテゴリに分類し，それらが直後の，2 回目から 5 回目までの発表内容に反映されていたかどうかを調べ，DRIP システムの有無で比較を行った。

5.5.3 実験結果と考察

実験期間中に行われたゼミの時間を表 5.1 に，ゼミごとに作成された会議コンテンツに含まれる発言数および議論セグメントの数を表 5.2 に示す。表 5.2 の括弧内の数字は議論セグメントの数を表している。これらの結果から，発表者 A と発表者 B のゼミにかかった時間や，ゼミ中に行われた発言数にはそれほど差がないことが分かる。

次に，2 回目以降のゼミ後に参加者に行ってもらったアンケートの結果を図 5.11 に示す（括弧付きの数字は前述した 5 つのカテゴリの番号と対応している）。これは，各カテゴリに分類された発言数の割合の推移を発表者別に示したものである。この結果から，(2) に分類された発言が DRIP システムを用いた発表者 A のゼミで

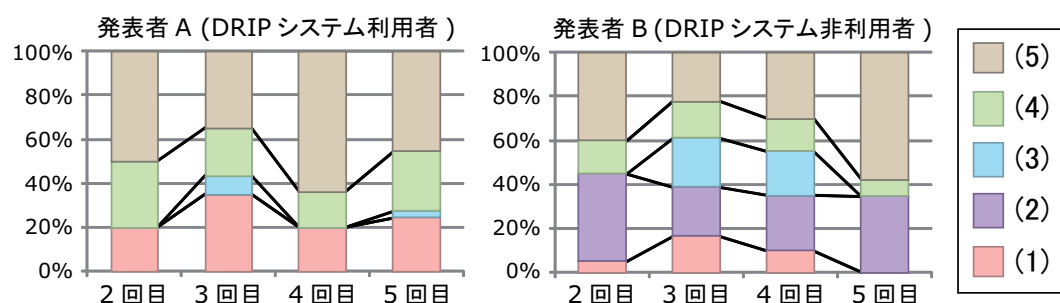


図 5.11: DRIP システムの有無による発表内容に関するアンケート結果の差異

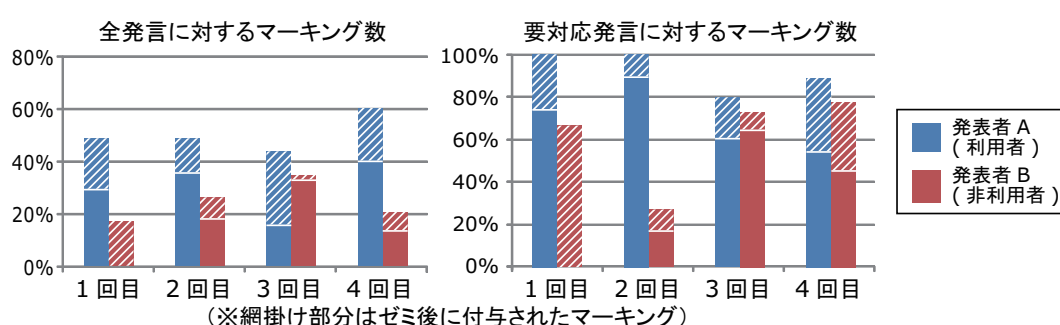


図 5.12: 全発言/要対応発言に対するマーキング数の比較

はまったく存在しないのに対して、DRIP システムを用いなかった発表者 B のゼミにおいて全体の 3 割近く存在していることが分かる。つまり、発表者 B は前回の議論を発表内容に十分に反映できていなかったことを示している。原因として発表者 B が Reflection に相当する作業を十分に行っていなかったため、反映すべき議論内容を誤って理解していた、もしくはその存在自体を忘れていたことが考えられる。

そこで、ゼミ中の全発言、および要対応発言に対して施されたマーキング数の割合（以下ではマーキング率と呼ぶ）を比較した結果を図 5.12 に示す。なお、網掛け部分は、ゼミ後に Discussion Browser を用いて入力されたものの割合を示しており、それ以外はゼミ中にポインタリモコンを用いて入力されたものの割合を表している。図 5.12 から、発表者 A の全発言へのマーキング率が発表者 B のそれをすべて上回っていることが分かる。これは、発表者 A は、DRIP システムを用いて Reflection を行う必要があるため、積極的にマーキングを施したのに対して、発表者 B は、DRIP システムを使用していないため、必要最小限のマーキングだけにとどまり、それ以上積極的にマーキングを施す動機づけが行われなかったと考えられる。このことは、発表者 B の 1 回目および 2 回目の要対応発言へのマーキング率が発表者 A のそれより低いことから推測できる（特に、1 回目に関して

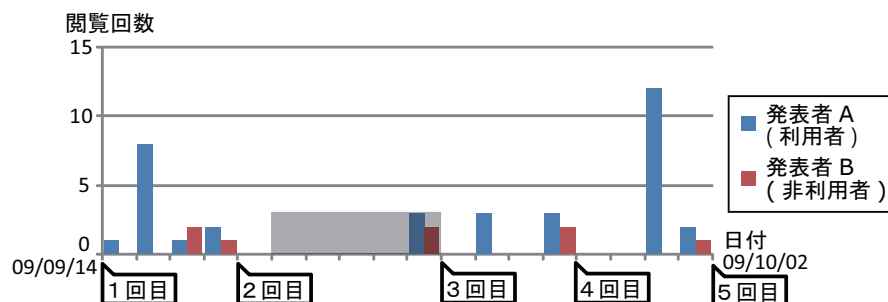


図 5.13: Discussion Browser における会議コンテンツの閲覧回数

表 5.3: 発表者 B の未対応発言に対するマーキング

対象回	カテゴリ (2)		カテゴリ (3)	
	あり	なし	あり	なし
1	5	3	0	0
2	1	3	2	2
3	3	2	4	0
4	7	2	0	0
計	16	10	6	2

は議論に集中していたために、ゼミ中のマーキングの入力を忘れていたと考えられる)。それに対して、要対応発言に対するマーキング率は、3回目および4回目では両発表者との間にそれほど差がないことが分かる。それにもかかわらず、図 5.11 において、発表者 B が発表者 A より過去の議論内容をふまえた発表ができていないのは、マーキング以降の Reflection が十分に行われていなかったからだと考えられる。次に、実験期間中における自身の発表に関する会議コンテンツの閲覧回数を時系列順に示したものを図 5.13 に示す。このグラフから、発表者 B がゼミの直前しか会議コンテンツを見直していないのに対して、発表者 A は複数日にわたって会議コンテンツを見直していることが分かる⁵。このことから、マーキングや Discussion Browser での閲覧だけを行っても十分な Reflection を行うことができなかったと考えられる。

次に、発表者 B のゼミでカテゴリ (2) もしくは (3) に分類された発言に対する、

⁵ 図 5.13 の網掛け部分は休日だったため、この期間における両発表者の閲覧回数が少なくなっている（研究室外のネットワークからのアクセスが遮断されていたため、自宅から会議コンテンツを閲覧することはできなかった）。この後の両発表者の 3 回目のゼミでカテゴリ (3) に該当する発言が存在することから、会議コンテンツを十分に見直すことができず、過去の会議コンテンツを十分に発表に反映できなかったことが推測される。

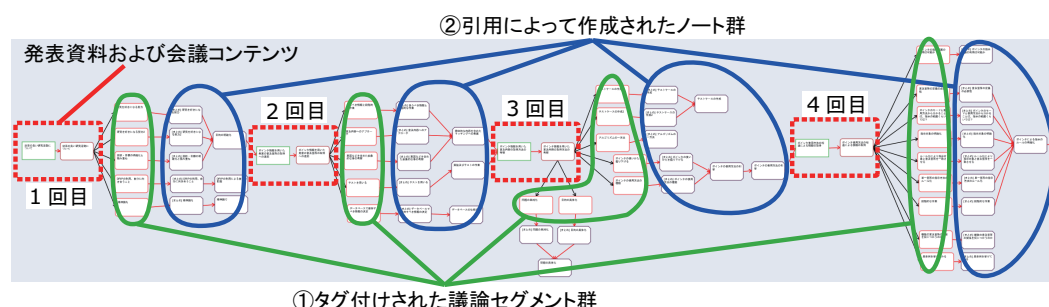


図 5.14: 実験期間中に DRIP システムを用いて記録されたコンテンツ間の関係

発表者 B のマーキング率を表 5.3 に示す。これから、発表者 B は（発表内容に反映された発言である (1) を除く）要対応発言に対して、ある程度マーキングを施しているにもかかわらず、その内容が十分に反映されていないことが分かる。前述のアンケートとは別に、実験後、参加者に対して 1 回目から 4 回目までの、カテゴリ (2) もしくは (3) に分類された発言をすべて見直してもらったところ、それぞれの発表において同じ内容について述べている発言があることが分かった。また、発表者 B はそのような発言に対してもマーキングを施していることが確認された。つまり、その重要性について認知はしていたが、それに対する回答を十分に考えていなかったと思われる。それに対して、発表者 A のゼミでは、カテゴリ (2) もしくは (3) に分類された発言がそれほど存在しないことから、DRIP システムの持つ、会議コンテンツを引用しながらアイデアや知識をノートとして記録する機能によって、発表者のアイデアや知識の顕在化が促された可能性がある。

図 5.14 は、発表者 A の DRIP システムの利用ログを可視化したものであり、5 回目のゼミを行う直前の時点の情報を表している。点線の枠で囲まれた箇所がゼミで使用された発表資料および会議コンテンツを表しており、そこからタグが付与された議論セグメント群 (①) が発生し、さらにそれらを引用したノート群 (②) が生まれ、それらが次のゼミへとつながっている様子が確認できる (図 5.15)。なお、発表者 A が DRIP システムで本実験期間中に付与したタグの数は 66 個（重複を除く）、コンテンツの引用によって生成されたリンク情報の数は 58、作成されたノートの数は 38 個であった。このように過去の会議コンテンツを引用しながらタスクの成果をノートとして記録していくことで、議論内容を十分に反映したタスクの遂行や発表ができ、その結果、図 5.11 のように (1) に該当する発言数の割合が DRIP システムを使わなかった発表者 B のものより全体的に多くなる結果につながったと思われる。

以上の結果より、DRIP システムを用いることによって、過去の議論内容を十分に反映した発表ができることを確認した。

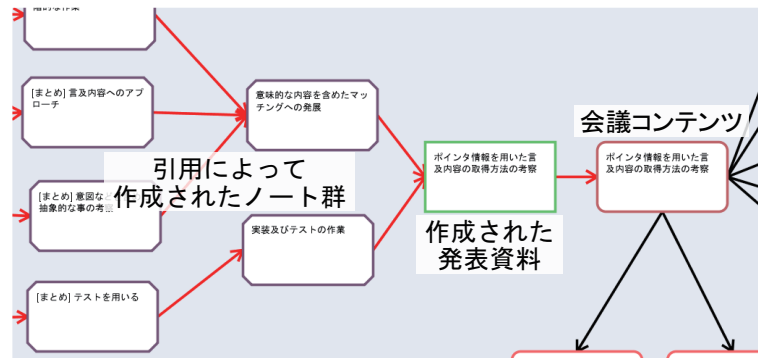


図 5.15: 引用によって作成されたノートから次のゼミへのつながり

5.6 本章のまとめ

本章では、議論フェーズで作成された会議コンテンツを再利用しながら、再認フェーズ、探求フェーズ、集約フェーズのそれぞれを支援する仕組みを備えた Discussion Mediator について述べた。まず、議論内容の効率的な閲覧を行うために再認フェーズでは、会議コンテンツ内の発言集合を引用しながら、自分なりのまとめの作成、およびその内容に合わせて TODO を作成する機能を実現した。そして、探究フェーズでは、会議コンテンツを閲覧しながら、新たなアイデアや知識を創造する。創造されたアイデアや知識はノートとして記録される。そして、集約フェーズでは蓄積された知識活動マップを利用することで次の発表に利用する発表資料の作成を行う。また、Discussion Mediator と前章で述べたアプリケーションを含めた DRIP システムの評価実験を行うことで、システムを利用している人間の知識活動が、過去の議論を適切に反映した、より密度の濃い効率的なものになっていることを確認した。

第6章 関連研究

本章では，議論内容の記録，議論の構造化・可視化，会議コンテンツの再利用の3点において本研究と関連する研究を挙げ，それらの研究との相違点について述べる．

6.1 議論内容の記録に関する研究

6.1.1 LiteMinutes

会議で行われた議論の様子を記録する研究は数多く行われている．Chiuらは LiteMinutes と呼ばれる Web ブラウザ上でマルチメディア議事録を作成するシステムの研究を行っている [1]．LiteMinutes は，カメラが設置された部屋の中でスライドを用いた議論を行い，議論の様子を映像・音声情報として記録する．また，書記が無線ラップトップ PC 上で専用のツールを用いることで，議論内容をテキストとして入力する．テキストの入力された時間は自動的に取得され，その時間情報を用いて関連するスライドや映像・音声情報と入力されたテキストとの間にリンク情報を付与する．

時間情報を用いて，関連するスライドや映像・音声情報とリンクする点では Discussion Recorder に類似する点が多い．しかし，これだけでは発言時間による正確な映像・音声情報のセグメンテーションを行うことができないという問題点がある．LiteMinutes では，メールによってセグメンテーション情報の修正を行っているが，Discussion Recorder では，デバイスを用いることで正確なセグメンテーション情報を暗黙的に取得できる．また，書記が入力したテキスト間には意味構造が付与されていないため，効率的な閲覧を行うことができないという問題点もある．

6.1.2 Portable Meeting Recorder

議論の様子を映像・音声情報として記録することによって，発言内容だけでは補えないノンバーバルな情報も合わせて閲覧できるため，より深い理解につながることを期待できる．しかし，一般に映像・音声情報は情報量が多いため，効率的に閲覧するためにはインデックス情報が必要となる．

映像・音声情報を用いて議論の様子を記録する研究例として Portable Meeting Recorder[10] がある．この研究では全方位カメラと4つのマイクを用いて記録を行う．4つのマイクによって発言者の方向の特定し，その方向の映像情報から人物抽出をする．このように音声処理や映像処理を組み合わせることで，映像・音声情報を効率的に閲覧するためのインデックス情報を自動的に付与する．同時にマイクによって取得された発言内容は，音声処理によってテキスト化され，先ほどのインデックス情報と共にメタデータとして XML 形式で記述される．

このようにして取得された映像・音声情報とメタデータは，MuVIE と呼ばれるインタフェースで閲覧できる．メタデータを利用することによって，特定の人物が行っている発言だけを閲覧したり，スライドやホワイトボードのビデオを閲覧したりする，といったようにユーザのさまざまな目的に合わせた閲覧方法を提供できる．

このシステムで作成されるインデックス情報は参加者の動作や音声のパワーなどを利用して自動的に取得されるため，作成のコストは Discussion Recorder より低い．また，これらの情報は意味内容を解釈した結果得られる情報ではないため，必ずしも閲覧者の目的に合致した閲覧方法を提供できるとは限らない．それに対して Discussion Browser は，Portable Meeting Recorder よりもコストは大きいですが，導入・継続という発言タイプによる議論のセグメンテーション情報，ポインタリモコンによる発言に対する賛成や反対の数のように，議論の文脈情報を考慮した閲覧インタフェースを提供できる点が異なる．

6.1.3 MinuteAid

議論の参加者がインタラクティブに議論の内容を編集する研究例として Lee らの MinuteAid[11] がある．MinuteAid は，Web サーバ・クライアント型のシステムであり，参加者は Intelligent Meeting Room と呼ばれる専用の環境の中で会議を行う．会議に使用されるスライド情報を Presentation Recorder と呼ばれるデバイスによって，議論の様子を Meeting Recorder と呼ばれるデバイスによって取得する．また，参加者はクライアント PC 内の Microsoft Word を用いて，議事録の編集を行う．その際，ユーザは必要に応じてリクエストを発行する．リクエストが発行されると，サーバはリクエストが発行された時間に対応するスライドの画像や音声情報が自動的に Word ファイルに挿入される．ユーザはそのファイルを自由に編集することで個人用の議事録の作成を行う．

個人用にカスタマイズされた議事録を作成するという点では，本研究と似た思想を持っている．しかし，知識活動において会議は繰り返し行われるものであり，議事録は会議後の活動において利用されるものである．この研究では，カスタマイズされた議事録を用いた活動の支援や次の発表に向けた発表資料の作成支援など知識活動への応用にまでは至っていない．

6.2 議論の構造化・可視化に関する研究

6.2.1 gIBIS

議論の内容を構造化・可視化することによって閲覧者の議論内容に対する理解を深めることが期待できる．その代表的な研究例として挙げられるのが，議論支援グループウェアの IBIS(Issue Based Information Systems) である．IBIS では，共同作業時の問題解決において，発言を Issue(問題)，Position(立場)，Argument(賛否) の 3 種類に分類し，それらの関係をリンクとするグラフ構造によって議論を構造化する．そして，IBIS をベースとして GUI による議論支援を行うシステムが gIBIS[3] である．gIBIS は議論構造を表すグラフを表示するブラウザ，グラフを構成するノードに関する情報を表示するノードインデックスウィンドウといったコンポーネントから構成されており，ユーザはこれらのコンポーネントを利用して直観的な操作で議論構造の編集を行う．

IBIS や gIBIS は議論を構造化し，それを可視化することにより，議論の進行の流れを把握しやすくしたという点で意義がある．しかし，議論で発生する発言すべてをこれらのモデルで提案している発言のタイプのいずれかに分類することが難しいという問題点がある．そのため，Discussion Recorder では，議論の構造化を議論のセグメンテーションと捉え，ユーザが発言時にポインタリモコンを用いて導入・継続のどちらかのタイプを選択するという手法を選択している．これにより，ユーザは，自分の発言によって話題が切り替わるかを判断するだけでよくなり，ユーザ側の負担を軽減できる．

また，作成されたグラフをインタラクティブに操作することによって，より深い議論内容の理解を行うことができると考えられるが，これらのシステムではそこまでの考察は行われていない．Discussion Browser では「導入」「継続」という発言タイプによるリンク情報の他に，書記が入力したリンク情報を組み合わせてグラフビューを提供している．そしてユーザは，検索バーやシークバーを併用しながら，グラフビューをインタラクティブに操作することによって，議論内容をより深く理解できるだろう．

6.2.2 Discussion Structure Visualizer

議事録に含まれる言語情報を用いて自動的に議論を構造化・可視化する研究として DSV(Discussion Structure Visualizer)[41] が挙げられる．DSV では，従来のテキスト処理による話題構造の抽出に関する研究のような話題の切れ目の同定だけを行うのではなく，話題と話題との関係まで考慮した構造化を目指している．

この研究では，話題の境界を決定するために，類義語の使用(言い換え)や同じ単語の繰り返しといった語彙的結束性を用いている．そして，結束度の低いとこ

るから順に話題の境界とみなす．その後，それぞれの話題の特徴ベクトルを求め，特徴ベクトル間の類似度を計算し，その値が閾値以上のものに関してはリンク情報を付与する．分割される話題の数やリンクの数はパラメータによって自由に変更できる．しかし，話題や話題間の関係の抽出をテキスト処理で行っているため，必ずしもその精度は高いものであるとは限らない．これに対し，Discussion Recorderでは，人間がさまざまなデバイスを用いて暗黙的にメタデータを入力するため，その精度はより高いものになる．また，ユーザが Discussion Browser 上で会議コンテンツを閲覧しながらアノテーションを行うことによって，より詳細な議論間の関係を抽出することも可能である．

6.3 会議コンテンツの再利用に関する研究

5章では，作成された会議コンテンツを再利用する手法について述べた．会議内容の記録・検索・閲覧を行うための研究は数多く行われているが，会議コンテンツそのものを知識活動の中で再利用する研究は多くない．

本橋らは，ミーティングシナプスと呼ばれる会議支援システムを提案している[42]．ミーティングシナプスには2つの特徴がある．以下はその引用である．

1点目は，会議中にホワイトボードの書き込み情報を TODO や会議の資料と結びつけることで，TODO から関係する議論内容を参照可能にする点である．2点目は，日常業務中に作成した文書とその文書の作成中に参考にした文書・Web ページの間の参照関係や，同じ話題の議論など会議中の書き込み同士の関係を保存し，後にその関係性を利用して現在の議論に関係する議論経緯や作業経緯を参照可能にする点である．

ホワイトボードの書き込みや日常業務中の文書同士の時間的・空間的・論理的近傍性を利用して関連付けを行い，登録された関係性は情報シナプス [29] と呼ばれる情報関係性管理基盤に保存される．そして，登録された関係性は会議アプリケーションを用いて過去の経緯を検索する際に利用される．

議論と日常業務の情報を関連付けることで，意思決定から業務実行・確認作業までの業務サイクルを支援するという点において，本研究との類似点は多い．しかし，「会議中には，日常業務中の作業結果や過去の会議の議論を確認する場面が発生する」「このように，会議に関係する情報を日常業務や次の会議で再利用する」と述べているように「過去の議論内容の正確な想起」を「再利用」と定義している点が本研究とは異なる．本研究における「再利用」とは，過去の議論内容を参照するだけでなく，その内容を引用しながら新たなコンテンツを作成する部分も含まれる．

ミーティングシナプスのように、会議に付随するコンテンツと会議記録を関連付けることによって、会議内容の参照を効率的に行うための研究としては、Geyerらの TeamSpace[8] や倉本らの ReSPoM[25] がある。以下では、システムによって取得している情報に関する違いを中心に述べる。

TeamSpace は、チームにおける協調作業を支援するためのシステムであり、協調作業において重要な要素（議題やアクション・アイテム）の、会議中における作成・編集時間を用いて、会議風景を記録した動画像のインデキシングを行っている。しかし、我々はアクション・アイテムのような結果が生まれるまでに行われる発言そのものを記録する必要があると考えている。たとえば、その場では技術的な問題などによって保留にされたアイディアが、かなり後になってから再び話題に上ることがある。しかし、作成されたアクション・アイテムには、そのアイディアに関する記録はないため、参照することができない。このように、長期間にわたるゼミコンテンツの利用を実現するためには、議論に関する詳細なメタデータを取得する必要がある。

ReSPoM は、参加者が議論中に作成したメモと、作成のきっかけとなった発言の間に存在する集約関係 (by-参照) を登録し、それを利用することで参照したい議論に関する音声・映像情報を検索参照するシステムである。Discussion Recorder では、発言への参照情報として、ポインタリモコンによるマーキング情報を利用している。マーキングを施すことは、ReSPoM のようにメモを作成するよりも参加者の負担が軽いため、議論に集中できる。また、このマーキング情報は前述の Discussion Browser 上でも入力できるため、議論中に見落とした発言に対する参照情報も付与できる。

第7章 結論

7.1 本論文のまとめ

本論文では、アイデアや知識を創造する知識活動に内在するさまざまなタスクの中でも特に議論に着目し、会議コンテンツの作成、およびその再利用を統括的にサポートすることで、個人の知識活動の支援を行う仕組みを提案し、それを具体的に実現した。

2章では、知識活動を議論中心の複数フェーズからなるサイクルとして捉え、DRIP サイクルと呼ばれるサイクルを提案した。DRIP サイクルは、議論を行うことで参加者からさまざまな意見やアドバイスを獲得する議論フェーズ、獲得した議論内容を後から効率的に再利用するために整理する再認フェーズ、整理された議論内容に基づいて新たなアイデアや知識を創造する探究フェーズ、蓄積されたアイデアや知識を再利用して次の発表で用いる発表資料の作成を行う集約フェーズから構成される。

本研究で議論の重要性を指摘する以前から、議論中におけるメモの記録などを通じて、議論内容を後で再利用しようという試みはさまざまなところで行われてきた。しかし、そのための明確な方法論やその支援ツールなどが開発され、運用されている例はほとんどない。本研究では、知識活動の中でも特に重要な役割を果たしている議論の内容を知識活動の中で有効に再利用するために、このような行為を再認フェーズという知識活動の重要な一つのフェーズとして実行することをこのサイクルの中で提案した。そして、このサイクルを統括的にサポートするDRIP システムを提案した。DRIP システムは、DRIP サイクル内の各フェーズに合わせたアプリケーション群から構成されている。

3章、4章では、議論フェーズにおいてDRIP システムが提供するいくつかのアプリケーションについて述べた。具体的には、会議コンテンツの作成を行うための Discussion Recorder、会議コンテンツの検索・閲覧を行うための Discussion Browser、議論中に会議コンテンツの可視化や回顧を行うことによって、議論の円滑化・活発化を支援するための Discussion Visualizer や Discussion Reminder を提案した。

Discussion Recorder は、環境に設置されたセンサや参加者が所持するデバイスを用いて、映像・音声や発言テキスト、スライド情報の他に、発言者情報や議論構造といったメタデータを組み合わせた会議コンテンツの作成を行うためのアプリ

ケーションである。Discussion Recorder によって取得したメタデータの評価実験を行うことによって、その妥当性を確認することができた。

Discussion Browser は、Discussion Recorder によって取得したメタデータの俯瞰や、内部構造の検索を行うことで、効率的に会議コンテンツの閲覧を行うことができるアプリケーションである。しかし、会議コンテンツの利用状況に関するアンケート調査を行うことによって、会議コンテンツの作成・検索・閲覧だけでは、知識活動を十分にサポートできていないことが明らかになった。

また、取得したメタデータをリアルタイムで可視化する Discussion Visualizer や、過去の会議コンテンツの回顧を行うための Discussion Reminder についても述べた。これらのアプリケーションの運用に基づく考察から、メタデータの可視化によって参加者がより積極的に議論に参加できること、会議コンテンツの回顧による知識の共有化がいかに困難であるかが分かった。

5 章では、議論フェーズで作成された会議コンテンツが、再認フェーズや探求フェーズ、集約フェーズでいかに再利用されるかについて述べた。複数の人間が一堂に会する議論フェーズとは異なり、システム利用者が個人用のスペースで作業を行うため、Web サーバ・クライアント型のアプリケーションを提案した。

再認フェーズにおける支援として、作成された会議コンテンツ内の発言を引用し、自分の解釈でノートを作成することで議論内容をまとめ、会議コンテンツやノートに対して分類するためのキーワードタグを付与する機能について述べた。探究フェーズでは、整理した会議コンテンツを参照しながら、新たに生まれたアイデアや知識をノートとして記録する。ノートを作成・編集する際に、閲覧していた会議コンテンツを引用することによって、会議コンテンツとノートとの間には意味的なリンク情報が暗黙的に付与されることを述べた。そして、活動の成果をより正確に他者に伝達し、より多くのフィードバックを獲得するため、集約フェーズでは、これまでに蓄積された会議コンテンツやノートを引用することによって、過去の議論を踏まえた発表資料の作成支援を行う。そして、再び議論フェーズで議論を行うことによって、新たなフィードバックを獲得できる。そのフィードバックを繰り返し取り入れていくことで知識活動は発展していく。

本研究では、議論を中心とする知識活動の支援を目指してきたが、その他の文献調査やシステムの実装といったさまざまなタスクも統合することによって、ユーザの知識活動をより統合的かつ効率的に再利用できる仕組みを提供できる。また、システム上で行われたノートの編集履歴や、ノートの作成・編集に利用したコンテンツに関する情報を処理することによって、次に行うべきことを教えてくれるシステムや、自分と関連する知識活動を行っているコミュニティを発見できるシステムといった応用システムの構築も可能になるだろう。

7.2 今後の課題

本論文では、DRIP サイクルの中で、会議コンテンツの作成および再利用を統括的にサポートすることで、個人の知識活動を支援するシステムを提案し、その妥当性・有効性に関する評価を行った。しかし、本論文で述べた評価方法は十分なものでないため、今後も継続的に評価を行っていく必要がある。

たとえば、5.5 節で述べた評価実験は、短期間における被験者の少ない小規模な実験であったため、今後は被験者の数を増やし、より長期間にわたって評価実験を継続する必要がある。また、今回の実験では、「について考えるべき」「するように実装するべき」といったように具体的な提案をとまなう発言が多かったが、なかには具体的な解決策を持っているにもかかわらず、発表者に考える機会を与えるために「もっとよく考えるべき」と発言する可能性がある。そして、発表者が発言者の持っている解決策とは別のものを提案した場合、「もっとよく考えるべき」という過去の議論内容をふまえているにもかかわらず、その発言者は5.5.2 項で述べたカテゴリ (2) に分類してしまう可能性がある。そのため、今後の実験ではカテゴリ (2) に分類する際の基準について、より深い考察が必要だと考えている。

継続的な評価実験以外の課題としては、DRIP システムに蓄積された情報を用いた会議中の議論支援が挙げられる。具体的な支援として以下のものが挙げられる。

- 現在行われている発表が過去のどの議論を踏まえて行われているかを提示することで、参加者の内容理解を促す
- 発表者のための補助として、発表資料と関連付けられたノートを発表者用インタフェースに表示する
- 参照関係情報を利用して、Discussion Reminder の検索をより効率的に行えるようにする

また、本論文では、集約フェーズで作成された発表資料を用いた発表を行うことで、過去の議論を踏まえた発表ができるようになることは確認したが、それによって議論が円滑化・活発化するかを評価していない。そのため、長期の運用を通じてデータの収集を行うことで評価を行う必要がある。

その他にも、手書きの図を取り込むなど、より直感的にノートを蓄積できるような仕組みや、表やグラフ、ビデオなどさまざまなメディアを含む発表資料を作成できるような仕組みの実現も検討していきたい。

7.3 将来の展望

本論文では、知識活動に内在するタスクの一つである議論に着目し、議論内容を獲得し、効率的に再利用するための仕組みについて述べた。本章では、本論文

で提案した知識活動支援システムをより有益なものにするための展望について述べる。

7.3.1 知識活動支援システムの拡張

本論文で述べた DRIP システムは、知識活動支援システムを基盤として、会議中心の DRIP サイクルを支援するためのシステムであり、会議以外のタスクを網羅的に支援するものではない。そのため今後、知識活動支援システムを拡張することによって、より網羅的な知識活動支援の実現が望まれる。

さまざまな種類の会議への対応

本論文では、大学研究室におけるゼミのように、自身がこれまでに行ってきた活動の内容を他者に向けて発表し、さらに自分だけでは解決できないような問題点について意見を出し合うような会議を対象に DRIP システムの構築を行った。しかし、われわれが普段口にしていく会議にはこのほかにもさまざまな種類が存在する。著者は、知識活動の中で行われる会議を分類するための指標として、会議の進行形式（型）や会議を行うための事前準備がどれだけ必要か、という観点があると考えている。ここでは、型や事前準備をあまり必要としない会議をカジュアルな会議、そうでない会議をフォーマルな会議と呼ぶことにする。カジュアルな会議の例としては、2, 3 名の人数がホワイトボードを用いて行うような議論や、4, 5 名程度のプロジェクトメンバーが進捗確認などを行うようなプロジェクトミーティングが挙げられる。それに対して、本研究で対象とした会議はフォーマルな会議に相当する¹。今後、これらすべての会議を統括的にサポートできるように、知識活動支援システムを拡張する必要があると考えている。

情報技術を用いて実世界インタラクションである会議の内容を記録する場合、その会議の種別（カジュアルかフォーマルか）によって考慮すべき点は、会議記録の設計（どのような情報を取得するか、どのように構造化するか）およびその記録方法である。フォーマルな会議であるほど、事前にプレゼンテーション資料を作成したり、大人数の参加者が対面して議論に参加したりするため、より密度の濃い議論、つまり再利用性の高い議論が行われる機会が多い。そのため Discussion Recorder では、さまざまなセンサやデバイスを用いて詳細な会議コンテンツの作成を行った。逆にカジュアルな会議では、特にプレゼンテーション資料を用意せず、気軽に議論を行っているため、その再利用性はフォーマルな会議に比べ低い

¹その会議がカジュアルであるかフォーマルであるかを判断する際、著者は、カジュアルな会議とフォーマルな会議を分類するような基準があるわけではなく、型や事前準備の必要性に基づいて相対的に決まるものである。

と思われる．そのような会議に Discussion Recorder のようなツールを用いることは，運用コストの面からも議論の進行の阻害という面からも適切ではない．

著者が所属する研究室では，よりカジュアルな会議の内容を記録するためのシステムに関する研究がすでに行われている．具体的には，ホワイトボードを用いたミーティング情報を獲得するシステム [16] や，共有ディスプレイを用いたプロジェクトミーティングに関する情報を獲得するシステム [15, 9] に関する研究が行われている．これらのシステムで作成されたコンテンツおよび Discussion Recorder で作成された会議コンテンツは独立したものではなく，相互に関連をもったコンテンツである．なぜならホワイトボードで描画された図がプロジェクトミーティングを通じてブラッシュアップされ，それがゼミで使用するプレゼンテーション資料に引用されるように，カジュアルな会議で生まれたコンテンツが，再利用性の高い状態でフォーマルな会議に利用されることがあるからである．

この一連の流れを，知識活動支援システムの中で紐付けながら蓄積することによって，最終的な成果物が生まれるまでに行われた議論，そしてその議論の中で採用されたアイディアや採用されなかったアイディアを芽づる式に引き出すことができるようになる．

会議以外のタスクのサポート

本論文では，知識活動のタスクの一つである会議に着目して DRIP システムを構築したが，今後は会議以外のさまざまなタスクを支援できるように知識活動支援システムを拡張していきたいと考えている．そのためには，タスクで作成・編集されるコンテンツに合わせて SyncNode と連携できるコンテンツサーバを作成し，引用が可能な RecNode を作成（もしくは既存の RecNode を拡張）する必要がある．

たとえば文献の調査や論文の執筆では，論文という活動内容をまとめたコンテンツが関わってくるため，論文を扱うようなコンテンツサーバを作成し，論文の部分要素を引用できるような RecNode を実現する必要がある．すでに著者が所属する研究室でも，論文執筆時に過去に閲覧した論文の部分要素を引用する仕組みを提案している [37]．この仕組みを知識活動支援システム上に実現し，さらには Discussion Mediator で作成したノートも合わせて引用できる仕組みを提供することで，より円滑な論文執筆支援を実現できると考えられる．このほかにも，論文以外にも，活動内容を分かりやすい形式で公表するという意味では，デモビデオも有効だと考えられるため，ビデオコンテンツにも対応することが望ましいだろう．

このようにさまざまなコンテンツに対応できるように知識活動支援システムを拡張することで，より複雑な知識活動マップを構築できる．それによって，これまでになかった新しいコンテンツ（いくつかのコンテンツをマッシュアップしたような複合コンテンツ）の作成や，よりユーザの知識活動に密着した支援を実現

することができる。

7.3.2 知識活動マップを用いた応用

ライフログとしての知識活動マップ

近年，ライフログのように人間の行動をデジタルデータとして保存し，それに基づいて追体験したり，記憶の想起を促したりするようなシステムに関する研究が広く行われている．代表的な例としては Fertig らの Lifestreams[6] や Gemmell らの MyLifeBits[7] が挙げられる．特に MyLifeBits では，読んだ本や聞いた CD だけでなく電話やメールの内容に至るまでさまざまなデジタルコンテンツの記録を目指している．

ライフログのように人間の活動を記録するという考え方は，知識活動支援システムにも応用できる．知識活動を行う過程の中には，成功しているものだけではなく，失敗したものも数多く存在するだろう．しかし，失敗や失敗の積み重ね自体が非常に重要な意味を持つこともある．別の見方をすれば，それらの失敗は当時扱っていたテーマにおいて「失敗」要因となっているだけであり，別のテーマから捉えなおしてみれば「成功」要因となる可能性もある．本研究では，このような失敗例も再利用できる仕組みの実現を視野に入れている．

知識活動マップには「そのコンテンツがどのような考えに基づいて作成されているのか」という文脈情報が引用情報という形で記録されている．この文脈情報を効果的に閲覧できるインタフェースを提供することによって，そのユーザの知識活動に関する背景知識の獲得を支援できる．活動を行っている本人が閲覧することで，現在行っていることの位置づけや疎かにしていることを確認できるため，よりよい活動を行うことが期待できる．また，第三者が知識活動マップを閲覧することによって，知識の伝承を効果的に行うことができる．特に後者の応用は，人材の流動化が進む現代社会にとって非常に重要な貢献をもたらすことが予想される．この応用例として，前述した Discussion Browser の拡張が挙げられる．これまでの Discussion Browser は単一の会議コンテンツしか閲覧することができなかったが，知識活動マップを利用して図 7.1 の左部のように会議コンテンツ間の関係を可視化することで，複数の会議コンテンツにまたがった閲覧を行うことができ，より深い議論内容の理解を実現できるだろう．

知識活動マップの分析に基づくコミュニティ発見

知識活動マップを可視化するだけでなく，演算を行うことで，さらなる知識活動支援を実現できる．たとえば，知識活動マップの構造の類似性を計算することで，自分と似た知識活動を行っている別の人間を発見し，新たなコミュニケーション

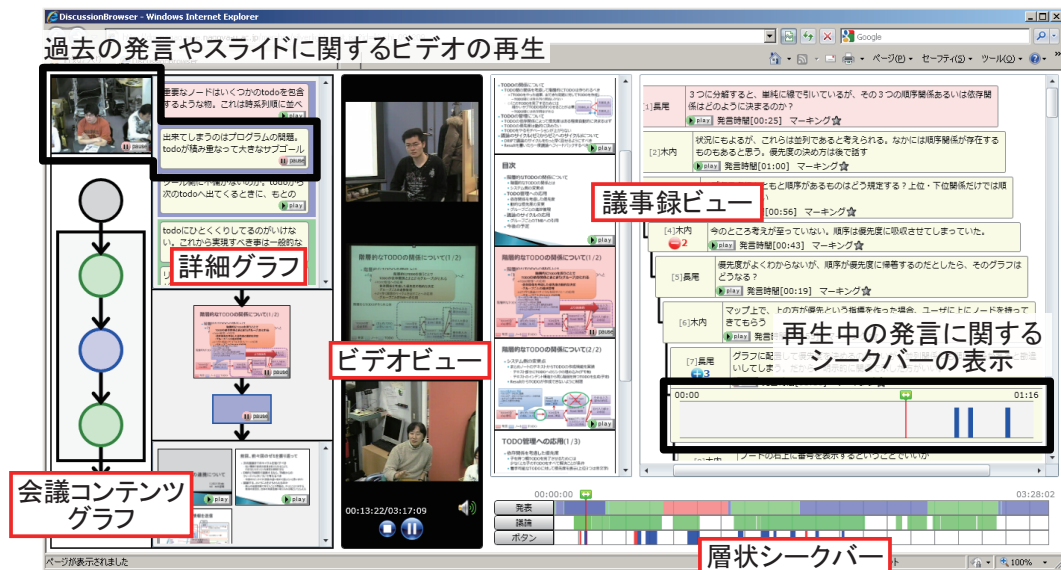


図 7.1: ディスカッションブラウザにおける会議コンテンツ間の関連性の可視化

ンを生み出すきっかけを与えることができる．また，マップ間の差分を計算することで，優れた成果を上げるための手法を知るための手掛かりを得られるかもしれない．このように，知識活動支援システムによって作成された知識活動マップはさまざまな可能性を秘めている．

7.3.3 グループの知識活動サイクルの支援

本論文で述べた知識活動支援システムの対象となる知識活動は，個人によって行われるものに限定されている．しかし，実際には個人だけでなく，複数の人間がグループを形成して知識活動を行うことも多い．今後はグループによる知識活動の支援にも取り組んでいく予定である．

グループで知識活動を行うためには，グループのメンバー全員がテーマに対する取り組み方やお互いの進捗状況を把握していることが必要になる．そのため，これまでのグループウェアという研究分野の考え方も取り入れながら知識活動支援システムの拡張を行う．その際，重要な点の一つとして，グループ内におけるコンテンツの共有が挙げられる．たとえば，参考になる文献を他のメンバーにも提示することでグループ全体で同じ知識を共有できる．また，グループのメンバーが作成した図的情報も Willustrator[21] のような仕組みを用いて，Web 上で（編集履歴を含めた）イラスト情報を共有することによって，新たなコンテンツを作成する際に再利用できる．

そして，知識活動支援システムを繰り返し利用することで，個人の知識活動マッ

プが作成されるように，グループの知識活動マップを作成することが考えられる．その際考慮しなければならない点は，メンバーの多様性である．同じグループ内でも担当している作業が異なれば，保有している知識の量や内容も異なる．そのため，ユーザごとに知識活動マップの閲覧方法や知識活動マップを用いた知識活動の支援方法を適合させる必要がある．

謝辞

本論文は、名古屋大学大学院情報科学研究科メディア科学専攻長尾研究室に在籍中の研究成果をまとめたものです。本研究を進めるにあたり、終始懇切な御指導・御助言を賜りました。ここに感謝の意を表します。

同専攻教授の長尾確先生には、研究内容に関する御助言だけでなく、研究者としての姿勢や研究活動そのものに関する御指導を賜りました。ここに深謝の意を表します。また、同専攻准教授の松原茂樹先生、並びに、同専攻客員教授の中岩浩巳先生には、副査として本論文の細部にわたり御助言を頂きました。ここに深謝の意を表します。

同専攻助教の大平茂輝先生には、研究内容に関する御助言だけでなく、本研究のシステム運用にあたり、多大な御助力を賜りました。厚く御礼申し上げます。オーマ株式会社の友部博教博士には、同研究室所属時にプロジェクトリーダーとして、研究内容に関する御助言を頂きました。ここに感謝の意を表します。

同大学工学研究科計算理工学専攻河口研究室助教の梶克彦博士、並びに、名古屋工業大学大学院工学研究科高橋・片山研究室助教の山本大介博士には、博士後期課程の先輩として、様々な御助言を賜りました。御礼申し上げます。また、小酒井一穂さんや石戸谷顕太郎さんをはじめとする諸先輩・同輩・諸後輩の皆様には、研究内容に関する議論に参加して頂いたり、研究生活に関するサポートをして頂いたりしました。ここに謝意を表します。そして、同研究室の秘書をされていた金子幸子さん、そして、現在の秘書をされている鈴木美苗さんには、研究活動を進めるにあたって様々なサポートをして頂きました。ここに謝意を表します。

参考文献

- [1] P. Chiu, J. Boreczky, A. Girgensohn, and D. Kimber. LiteMinutes: An Internet-Based System for Multimedia Minutes. In *Proc. 10th World Wide Web Conference (WWW 2001)*, pp. 140–149, 2001.
- [2] P. Chiu, A. Kapuskar, S. Reitmeier, and L. Wilcox. Room with a Rear View: Meeting Capture in a Multimedia Conference Room. *IEEE MultiMedia*, Vol. 7, No. 4, pp. 48–54, 2000.
- [3] J. Conklin and M.L. Begeman. gIBIS: A Hypertext Tool for Exploratory Policy Discussion. In *Proc. 1988 ACM Conference on Computer Supported Cooperative Work (CSCW 1988)*, pp. 140–152, 1988.
- [4] G. Cruz and R. Hill. Capturing and playing multimedia events with STREAMS. In *Proc. 2nd ACM International Conference on Multimedia (MULTIMEDIA 1994)*, pp. 193–200. ACM, 1994.
- [5] Thomas H. Davenport. *Thinking for a Living: How to Get Better Performances and Results from Knowledge Workers*. Harvard Business School Press, 2005. (邦訳:『ナレッジワーカー』 藤堂圭太訳 , ランダムハウス講談社 , 2006).
- [6] S. Fertig, E. Freeman, and D. Gelernter. Lifestreams: An Alternative to the Desktop Metaphor. In *Proc. ACM SIGCHI Conference on Human Factors in Computing Systems (CHI 1996)*, pp. 410–411, 1996.
- [7] J. Gemmell, G. Bell, R. Lueder, S. Drucker, and C. Wong. MyLifeBits: Fulfilling the Memex Vision. In *Proc. 10th ACM International Conference on Multimedia (MULTIMEDIA 2002)*, pp. 235–238, 2002.
- [8] W. Geyer, H. Richter, and G. D. Abowd. Towards a Smarter Meeting Record—Capture and Access of Meetings Revisited. *Multimedia Tools and Applications*, Vol. 27, No. 3, pp. 393–410, 2005.
- [9] K. Ishitoya, S. Ohira, and K. Nagao. TimeMachineBoard: A Casual Meeting System Capable of Reusing Previous Discussions. In *Proc. 5th International Conference on Collaboration Technologies (CollabTech 2009)*, pp. 84–89, 2009.

- [10] D. Lee, B. Erol, J. Graham, J. Hull, and N. Murata. Portable Meeting Recorder. In *Proc. 10th ACM International Conference on Multimedia (MULTIMEDIA 2002)*, pp. 493–502, 2002.
- [11] D. Lee, J.J. Hull, B. Erol, and J. Graham. MinuteAid: Multimedia Note-Taking in an Intelligent Meeting Room. In *Proc. IEEE International Conference on Multimedia & Expo (ICME 2004)*, Vol. 3, pp. 1759–1762, 2004.
- [12] K. Nagao. *Digital Content Annotation and Transcoding*. Artech House Publishers, 2003.
- [13] K. Nagao, K. Kaji, D. Yamamoto, and H. Tomobe. Discussion Mining: Annotation-Based Knowledge Discovery from Real World Activities. In *Proc. 5th Pacific-Rim Conference on Multimedia (PCM 2004)*, pp. 522–531, 2004.
- [14] I. Nonaka and H. Takeuchi. *The Knowledge Creating Company: How Japanese Companies Create the Dynamics of Innovation*. Oxford University Press, 1995.
- [15] 石戸谷顕太郎, 小幡耕大, 大平茂輝, 長尾確. カジュアルミーティングにおける議論コンテンツの構造化とその応用. 情報処理学会第 71 回全国大会講演論文集, pp. 4.55–4.56, 2009.
- [16] 伊藤周, 土田貴裕, 大平茂輝, 長尾確. 創造的議論の再利用を促進するカジュアルミーティングシステム. 情報処理学会第 70 回全国大会講演論文集, pp. 4.377–4.378, 2008.
- [17] 大江展弘, 平野靖, 梶田将司, 間瀬健二. 全方位画像からのインタラクション認識を用いた会議記録の編集・閲覧システム. 情報処理学会第 67 回全国大会講演論文集, pp. 4.149–4.150, 2005.
- [18] 大須賀節雄. データベースと知識ベース -新しい情報システムを目指して-. オーム社, 1989.
- [19] 梶克彦, 長尾確. Annphony: メタコンテンツ処理のためのプラットフォーム. 情報科学技術フォーラム (FIT)2006, 第 5 巻, pp. 381–384, 2006.
- [20] 梶克彦, 長尾確. 部分引用の管理に基づく web コンテンツのマッシュアップ. 情報処理学会第 69 回全国大会講演論文集, pp. 1.407–1.408, 2007.
- [21] 神原啓介, 安村通晃. Willustrator: Web 上での協同イラストレーション. 第 13 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 (WISS 2005), pp. 139–140, 2005.

- [22] 國藤進. ナレッジマネジメントによる“知の共鳴”. 情報処理, Vol. 47, No. 9, pp. 1021–1027, 2006.
- [23] 久保田秀和, 齊藤憲, 角康之, 西田豊明. 会話量子化器を用いた会話場面の記録. 情報処理学会論文誌, Vol. 48, No. 12, pp. 3703–3714, 2007.
- [24] 久米均. シリーズ 現代工学入門 品質管理. 岩波書店, 2005.
- [25] 倉本到, 野田潤, 藤本典幸, 萩原兼一. 会合における備忘録をもとに一次記録を検索参照する会合情報記録検索システム ReSPoM. 情報処理学会論文誌, Vol. 41, No. 10, pp. 2804–2813, 2000.
- [26] 栗原崇, 井上直人, 葛岡英明, 鈴木栄幸. 遠隔ミーティングの記録・再生を支援するシステムの研究. 情報処理研究報告, No. 2005-GN-56, pp. 31–36, 2005.
- [27] 小林賢治. 知識継承ソフトウェア KnowledgeMeisterSucceedTM. 東芝レビュー, Vol. 61, No. 7, 2006.
- [28] 齋藤孝. 会議革命. PHP 研究所, 2002.
- [29] 坂上秀和, 本橋洋介, 坂口基彦, 内藤圭三. 情報間の関係性管理に基づく企業内情報構造化方式 情報シナプスの構想. 情報処理研究報告, No. 2009-GN-070, pp. 43–48, 2009.
- [30] 佐土原健, 李時旭, 児島宏明. サブワードユニット上の文字列カーネルを用いたタスク非依存なトピック分割. 人工知能学会第 21 回全国大会講演論文集, No. 3C9-4, 2007.
- [31] 島津秀雄, 小池晋一. KM 再考: Web2.0 時代のナレッジマネジメント. 情報処理, Vol. 47, No. 7, pp. 768–774, 2006.
- [32] 角康之, 西田豊明, 坊農真弓, 來嶋宏幸. IMADE: 会話の構造理解とコンテンツ化のための実世界インタラクション研究基盤. 情報処理, Vol. 49, No. 8, pp. 945–949, 2008.
- [33] 角康之, 間瀬健二. インタラクション・コーパス構築の試みとしてのミーティングキャプチャ. ヒューマンインタフェースシンポジウム 2002, pp. 241–244, 2002.
- [34] 土田貴裕, 大平茂輝, 長尾確. 対面式会議コンテンツの作成と議論中におけるメタデータの可視化. 情報処理学会論文誌, Vol. 51, No. 2, pp. 404–416, 2010.

- [35] 友部博教, 土田貴裕, 大平茂輝, 長尾確. ディスカッションメディア: 会議コンテンツの構造化と効率的な閲覧システム. 人工知能学会第 21 回全国大会論文集, No. 2F3-5, 2007.
- [36] 友部博教, 長尾確. ディスカッションマイニング: 議事録集合からの知識発見. 情報処理学会第 67 回全国大会講演論文集, pp. 4_65–4_66, 2005.
- [37] 林亮介, 土田貴裕, 大平茂輝, 長尾確. 閲覧時アノテーションを利用した Web ドキュメントの引用とその応用. 情報処理学会第 70 回全国大会講演論文集, pp. 1_535–1_536, 2008.
- [38] 平島大志朗, 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. 情報処理研究報告, No. 2005-GN-55, pp. 81–86, 2005.
- [39] 古田一雄, 前原基芳, 高島亮祐, 中田圭一. 知的支援機能を備えた電子会議システム. 社会技術研究論文集, Vol. 1, pp. 299–306, 2003.
- [40] 堀公俊, 加藤彰. ファシリテーション・グラフィック. 日本経済新聞出版社, 2006.
- [41] 松村真宏, 加藤優, 大澤幸生, 石塚満. 議論構造の可視化による論点の発見と理解. 日本ファジィ学会誌, Vol. 15, No. 5, pp. 554–564, 2003.
- [42] 本橋洋介, 坂上秀和, 内藤圭三. 会議支援システムを用いた会議内容情報の再利用性の考察. 情報処理研究報告, No. 2009-GN-72, pp. 1–6, 2009.
- [43] 吉田新一郎. 会議の技法. 中公新書, 2000.

研究業績

雑誌論文

- 土田貴裕，大平茂輝，長尾確．ゼミコンテンツの再利用に基づく研究活動支援．情報処理学会論文誌，Vol.51, No.6, pp.1357–1370, 2010.6.
- 土田貴裕，大平茂輝，長尾確．対面式会議コンテンツの作成と議論中におけるメタデータの可視化．情報処理学会論文誌 特集「インタラクションの基盤技術，デザインおよび応用」，Vol.51, No.2, pp.404–416, 2010.2 .

国際会議

- T. Tsuchida, K. Kiuchi, S. Ohira, and K. Nagao. Visualization of Discussions in Face-to-Face Meetings, The Fifth International Conference on Collaboration Technologies(CollabTech09), 2009.8.
- T. Tsuchida, S. Ohira, and K. Nagao. Knowledge Activity Support System Based on Discussion Content, The Fourth International Conference on Collaboration Technologies(CollabTech08), 2008.8.

研究会

- 土田貴裕，大平茂輝，長尾確．知識活動支援システムによる会議コンテンツ間の関連性の獲得とその応用．情報処理学会第 75 回グループウェアとネットワークサービス研究会，Vol.2010-GN-75, No.7, 2010.3.
- 土田貴裕，大平茂輝，長尾確．会議コンテンツの再利用に基づくプレゼンテーション作成支援．情報処理学会第 68 回グループウェアとネットワークサービス研究会，Vol.2008, No.48, pp.85–90, 2008.5.

その他の発表

- 土田貴裕，大平茂輝，長尾確．知識活動支援システムによる会議コンテンツ間の関連性の獲得とその応用．情報処理学会第 72 回全国大会，2010.3.
- 土田貴裕，友部博教，大平茂輝，長尾確．会議コンテンツの効率的な再利用に基づく知識活動支援システム．人工知能学会第 21 回全国大会，2007.6.
- 土田貴裕，友部博教，大平茂輝，長尾確．会議コンテンツの効率的な再利用に基づく知識活動支援システム．情報処理学会第 69 回全国大会，2007.3.
- 土田貴裕，友部博教，大平茂輝，長尾確．議事録に基づく知識活動サイクルの活性化．人工知能学会第 20 回全国大会，2006.6.
- 土田貴裕，友部博教，大平茂輝，長尾確．議事録に基づく知識活動サイクルの活性化．情報処理学会第 68 回全国大会，2006.3.
- 土田貴裕，長尾確．個人用知的移動体におけるヒューマントレーサの実装．情報処理学会第 67 回全国大会，2005.3.

その他の発表（共著）

- 木内啓輔，清水元規，土田貴裕，大平茂輝，長尾確．知識活動支援システムにおけるタスク管理と文脈情報の可視化．情報処理学会第 72 回全国大会，2010.3.
- 清水元規，木内啓輔，土田貴裕，大平茂輝，長尾確．ディスカッションマイニングにおける発言中の指示対象の取得とその応用．情報処理学会第 72 回全国大会，2010.3.
- 木内啓輔，土田貴裕，大平茂輝，長尾確．議事リマインダによる会議における議論の検索と利用．情報処理学会第 71 回全国大会，2009.3.
- 林亮介，土田貴裕，大平茂輝，長尾確．閲覧時アノテーションを利用した Web ドキュメントの引用とその応用．情報処理学会第 70 回全国大会，2008.3.
- 伊藤周，土田貴裕，大平茂輝，長尾確．創造的議論の再利用を促進するカジュアルミーティングシステム．情報処理学会第 70 回全国大会，2008.3.
- 林亮介，土田貴裕，大平茂輝，長尾確．論文の作成と再利用に基づく研究活動支援システム．人工知能学会第 21 回全国大会，2007.6.

- 伊藤周，土田貴裕，大平茂輝，長尾確．議論を誘発するカジュアルミーティング支援．人工知能学会第 21 回全国大会, 2007.6.
- 友部博教，土田貴裕，伊藤周，林亮介，成田一生，大平茂輝，長尾確．ディスカッションメディア：会議コンテンツの構造化と効率的な閲覧システム．情報処理学会第 69 回全国大会, 2007.3.

受賞

- 2008 年 5 月．情報処理学会第 68 回グループウェアとネットワークサービス研究会において優秀発表賞．
- 2005 年 3 月．情報処理学会第 67 回全国大会において学生奨励賞．