マイクロタスク型クラウドソーシングを用いた大規模データ処理における精度向上手法ならびにシステム開発と運用に関する研究

芦川 将之

電気通信大学大学院情報システム学研究科

博士（工学）の学位申請論文

2017年3月

マイクロタスク型クラウドソーシングを用いた大規模データ処理における精度向上手法ならびにシステム開発と運用に関する研究

博士論文審査委員会

主査　　大須賀 昭彦　教授

委員　　南 泰浩　教授

委員　　植野 真臣　教授

委員　　田原 康之　准教授

委員　　川村 隆浩　客員准教授

Studies on Quality Improvement of Large-scale Data Analysis Using Microtask Crowdsourcing and Practical System Deployment

Masayuki Ashikawa

Abstract

Crowdsourcing is an outsourcing service in which many tasks are processed by many unspecified people, and it is used in various domains for such purposes as analyzing and compiling large data. The number of workers that process crowdsourcing tasks is increasing in line with the expansion of the domains in which crowdsourcing is used. Therefore, the way in which work is performed in crowdsourcing is expected to become common practice. However, support for crowdsourcing workers, such as education and improvement of the working environment, is insufficient. This problem is thought to be due to crowdsourcing workers being numerous and unspecified. Crowdsourcing workers are employed and terminated easily since they are unspecified. This poor management of workers could lead to declining quality of workers records and unjustified termination of workers.

If workers produce high-quality results of tasks, there is no necessity for requesters to terminate workers. However, management and education for crowdsourcing workers are subject to several problems. For example, it is difficult to individually educate each worker, because the workers are numerous and unspecified in microtask crowdsourcing. In addition, personalized management and education for crowdsourcing workers undermines the merits of microtask crowdsourcing such as low cost and rapidity. In order to avoid easy dismissal, worker management and education can be viewed as useful methods. Therefore, we propose four worker selection methods and a grade-based training method.

However, for development of four worker selection methods and a grade-based training method for existing crowdsourcing services, the current worker control and task allocation

methods are insufficient and it is difficult to incorporate new worker selection methods and training method mechanisms in them. Therefore, we developed the Private Crowdsourcing System (PCSS). PCSS has been in operation since 2011. The number of PCSS workers is currently 2454, whereas the number of processed tasks is 18.5 million.

The four worker selection methods consist of preprocessing filtering, real-time filtering, post-processing filtering, and guess-processing filtering. In addition to a basic approach involving initial training or the use of gold-standard data, these methods include a novel approach, utilizing collaborative filtering techniques.

We also propose a grade-based training method that automatically allocates pre-learning tasks to the workers based on the concept proposed. In this method, each worker are instructed to process the pre-learning tasks prior to processing difficult tasks. Worker skill is upgraded by processing the pre-learning tasks. In particular, our system allocates appropriate pre-learning tasks by analyzing the correlations between tasks based on workers' records for 18.5 million tasks using a Bayesian network.

We also collected a large amount of vocabulary data for natural language processing, such as voice recognition and text to speech by using PCSS. We collected 517 million pages with a crawler. These pages include 319 million Japanese pages and 12.5billion Japanese sentences. Finally, we got 138 thousand vocabulary data.

The quality control methods increased accuracy 32.4 points in collecting vocabulary tasks. Furthermore, the grade-based training method automatically allocated 31 pre-learning task categories for 9 target task categories, and after the training of the pre-learning tasks, we confirmed that the accuracy of the target tasks was raised by 7.8 points on average.

Therefore, by combining the filtering methods and the training method, task requesters in microtask crowdsourcing can obtain higher-quality results without dismissing valuable workers.

マイクロタスク型クラウドソーシングにおける精度向上手法に関する研究

芦川 将之

概要

クラウドソーシングは Crowd(群衆) + Sourcing(調達) の造語であり,「企業，組織が，自社もしくはアウトソースの人材により実施していた業務を，よりオープンかつ不特定多数の Crowd(群衆) から人材を集め実施すること」と定義されている．このようなクラウドソーシング技術は，大規模データの解析や構築などを低コストで行うことが可能であり様々な分野や用途で利用されている．しかしその特性上，処理速度の速さや低コストの利点に対して処理結果の精度においては専門家による処理よりも劣るため問題視されており，様々な精度向上手法が研究されている．それらの研究では作業 (タスク) を処理する作業者 (ワーカー) が不特定多数ということもあり，安易に低品質なワーカーを排除する傾向がある．

しかし，その利用範囲の拡大に従いワーカーの数も増大しており，将来的にクラウドソーシングにおける作業が社会における一つの就労形態となることが予想される．そのような傾向にあるにもかかわらず，現状のクラウドソーシングではワーカーに対する安易な排除が中心となり，育成や労働環境の改善と言ったサポートが十分であるとは言い難い．これらの問題はクラウドソーシング市場自体の縮小にもつながりかねない．これらの問題に対応するためには，クラウドソーシング運用において通常の労働環境と同様に人材 (ワーカー) のマネジメントや育成が重要になってくると予想される．

我々はこのようなクラウドソーシングの精度問題において，ワーカーのフィルタリングと教育の二つの手法の組み合わせで対応を行っている．ワーカーのフィルタリングで適材適所な作業環境を用意し，その上で低品質なワーカーを高品質なワーカーへと成長させるべく教育を行う．

しかし従来のクラウドソーシングサービスでは我々の提唱するフィルタリングや段階的教育を実現するには外部のサービスが提供している機能の範囲では十分ではなく，外部のサービスに新規の機能を追加することも難しいという問題がある．我々はこれらの問題を

解決するために，独自のクラウドソーシングシステム (PCSS) を構築し，システム内にて精度向上手法を適用することで問題の解決を試みている．PCSS は 2011 年から運用を継続しており，1853 万個のタスクを処理した実績を持っている．

PCSS におけるワーカーのフィルタリングは事前フィルタリング，動的フィルタリング，結果フィルタリング，推測フィルタリングという 4 つの独自のフィルタリングの組み合わせで行われている．その過程でワーカーの各タスクに対する特性の解析を行い，適したタスクのアロケーション，または不適なタスクからの排除などを行う．また，その過程で低品質であることが判明したワーカーに対し，ワーカーがタスクを処理する過程で適切な学習タスクをこなすことで能力を向上させる段階的な学習方式を提案する．このような段階的な学習法式としては，学習支援システム (Intelligent tutoring system，ITS) における学習モデルをベイジアンネットワークによって表現する研究 [Ueno 00] が提案されており，その有効性が示されている．我々はこのベイジアンネットワークを用いた段階的学習手法のマイクロタスク型クラウドソーシングへの適用を提案する．具体的な手法として，まずワーカーのタスク処理結果からベイジアンネットワークを用いてタスク間の関係性の解析を行う．次にタスクを処理することで段階的な学習が可能となるような学習タスクを自動生成する．これによってワーカーの能力の育成を狙う．

さらに，これらのフィルタリングを実装した PCSS を用いて知識処理研究に必要な語彙の収集を行った．Web クローラを用いて 5.2 億ページの Web データの収集を行い，そこから形態素解析で得られた語彙候補に対して PCSS でノイズ除去，読み仮名などのデータ付与を行なうことで 14 万語の未知語を得ることに成功した．

この語彙収集の課程で行ったクラウドソーシング処理において，ワーカーのフィルタリングを行うことにより精度が 32.4 ポイント上昇していることを確認した．また，同様に低品質な結果の多いタスクに対して学習タスクの算出を行ったところ，9 種類のタスクに対して合計 31 種類の学習タスクを導出することが出来た．また，この導出された学習タスクを用いて低品質なワーカーに学習させ，改善効果を測定したところ平均 7.8 ポイントの改善効果が確認できた．比較対象として決定木でも学習タスクを導出したが，ベイジアンネットワークを用いて導出した学習タスクよりも効果が低いことが確認できた．

このようにクラウドソーシングにおいても適切なワーカーマネジメントと育成を行うこ

とで，安易にワーカーを排除すること無く高精度なデータ処理結果を高速かつ低コストで取得することが可能であることを示すことが出来た.

目 次

第 1 章　序論 **1**

第 2 章　関連研究 **5**

2.1　クラウドソーシングの分類 5

2.2　クラウドソーシングの利用に関する研究 8

2.3　クラウドソーシングの精度向上に関する研究 13

2.4　ICT を教育に用いた研究 15

2.5　クラウドソーシングや教育に機械学習を用いた研究 17

第 3 章　プライベートクラウドソーシングシステムの構築 **21**

3.1　精度向上手法の組み込みが可能なクラウドソーシングシステム 21

3.2　PCSS 上で処理される作業の分類 22

3.3　PCSS の詳細と運用 23

3.4　PCSS 上で作業するワーカーの特徴 35

3.5　PCSS と既存のサービスとの比較 37

第 4 章　ワーカーのフィルタリングによる精度向上手法の提案 **39**

4.1　ワーカーを対象とした精度向上手法 39

4.2　事前フィルタリング 40

4.3　動的フィルタリング 43

4.4　結果フィルタリング 47

4.5　推測フィルタリング 49

第 5 章　ワーカーのフィルタリングによる精度向上手法の評価及び考察 **55**

5.1 事前フィルタリングの効果 55
5.2 動的フィルタリングの効果 55
5.3 結果フィルタリングの効果 56
5.4 推測フィルタリングの効果 57
5.5 考察 61

第 6 章 ワーカーの段階的学習による精度向上手法の提案 63
6.1 クラウドソーシングにおける学習の必要性 63
6.2 段階的学習法の提案 66
6.3 STEP1: タスクグループのカテゴリ分類 68
6.4 STEP2: タスクカテゴリ間の関係性の解析 70

第 7 章 ワーカーの段階的学習による精度向上手法の評価及び考察 79
7.1 学習の有無による各ワーカーの精度向上の実験 79
7.2 学習の有無による各ワーカーの精度向上結果の評価 80
7.3 学習の有無による各ワーカーの精度向上結果の考察 84

第 8 章 ワーカーのフィルタリング及び段階的学習の事例紹介 93
8.1 語彙の重要性 94
8.2 クローリングによるテキスト収集 96
8.3 未知語候補の抽出 96
8.4 単語判定と単語情報付与 97
8.5 結果 101

第 9 章 結論 103
9.1 まとめ 103
9.2 今後の課題 105

謝辞 109

参考文献 111

研究業績 **123**

図 目 次

1.1 クラウドソーシング概要 1

2.1 クラウドソーシングの分類 6
2.2 クラウドソーシング市場規模推移予測（2011〜2017 年度）[Yano 13] . . . 8
2.3 クラウドソーシング関連の論文数の推移 9

3.1 PCSS におけるタスク 22
3.2 PCSS におけるタスク例 23
3.3 ポイント業者を経由したクラウドソーシング 24
3.4 PCSS のシステム構成 25
3.5 ワーカー視点での PCSS における処理の流れ 26
3.6 タスク選択フェーズ 27
3.7 トレーニングフェーズ 28
3.8 タスク処理フェーズ 29
3.9 リクエスタ視点での PCSS における処理 30
3.10 タスク登録ツール 30
3.11 クラウドソーシング API の位置付け 32
3.12 クラウドソーシング API の例 33
3.13 ワーカーの男女比 35
3.14 ワーカーの年齢分布 36
3.15 ワーカーの居住地比率 36
3.16 比較のための人物画像判定タスク 38

4.1 PCSS におけるワーカーに対する精度向上手法 40

4.2 事前フィルタリング 41
4.3 アクセント能力者を優先させるためのテスト例 41
4.4 多数決タスクの例 44
4.5 ワーカーに表示されるステータス画面 45
4.6 タスク処理結果を用いたワーカーの特徴付け 48
4.7 推測フィルタリング 49
4.8 PCSS におけるフィルタリングの組み合わせ 54

5.1 実測タスク結果精度 $M_{u,i}$ と予測タスク結果精度 $P_{u,i}$ の比較 59

6.1 タスクカテゴリごとのワーカーの精度の相関性（一部） 64
6.2 低品質ワーカーが与える悪影響 65
6.3 学習タスクカテゴリ導出のためのステップ 68
6.4 ベイジアンネットワークをクラウドソーシングに用いた例 71
6.5 精度改善対象タスクカテゴリにおける有向グラフの例 74
6.6 TID0: 読点の位置が正しいか判定 75
6.7 TID0 に対する学習タスク 75
6.8 ベイジアンネットワークを決定木に用いた例 76

7.1 段階的学習手法の効果を確認するための実験 81
7.2 精度改善対象タスクカテゴリ TID1 におけるワーカーの成長パターン . . . 90
7.3 小規模データから得られた有向グラフ 91

8.1 語彙抽出フロー 93
8.2 電子書籍読み上げにおける読み誤り原因 94
8.3 Web からの新語抽出フロー 95
8.4 単語判定タスク 97
8.5 品詞付与タスク 98
8.6 読み付与タスク 99
8.7 アクセント付与タスク 100

8.8 コスト削減効果 102

9.1 ハイブリッドクラウドソーシング 107

表 目 次

3.1 クラウドソーシング API オペレーション 34
3.2 PCSS の運用実績 34
3.3 高ランクワーカー 37
3.4 Amazon Mechanical Turk と PCSS との精度比較 38

4.1 事前フィルタリングによるベースフィルタリング 42
4.2 タスク別ワーカー結果精度（一部） 46
4.3 各カテゴリにおける値 47
4.4 コンテンツベースの協調フィルタリングのデータ例（「-」部分は未作業） . 51
4.5 アイテムベースの協調フィルタリングのデータ例（「-」部分は未作業） . . 52
4.6 ワーカー間類似度（一部） 53

5.1 各カテゴリにおける値 56
5.2 「スキル保持」「負スキル保持」と判定されたワーカ数 56
5.3 実測タスク精度と予測タスク精度の比較 58
5.4 各カテゴリにおける精度向上効果 58
5.5 各カテゴリにおけるワーカー数 60

6.1 特定のタスクで精度が悪いワーカーの例 64
6.2 タスク間類似度 70
6.3 タスクカテゴリごとのタスク処理結果精度（一部） 73
6.4 精度改善タスクカテゴリ一覧 74
6.5 精度改善タスクカテゴリと対応する学習タスクカテゴリ 78

7.1 学習タスクカテゴリ実施の有無によるタスク改善効果（ベイジアンネットワーク） 83
7.2 学習タスクカテゴリ実施の有無によるタスク改善効果（決定木） 84
7.3 小規模データにおける精度改善タスクカテゴリ一覧 91
7.4 小規模データから得られた学習タスク実施の有無によるタスク改善効果 . . 92

8.1 獲得したWebテキスト 96
8.2 各タスクの作業結果における一致率 101
8.3 未知語獲得数 101

第1章　序論

本章では，本研究の背景を述べた後，本研究の目的と貢献を説明する．その後，本研究の構成について述べる．

クラウドソーシングはCrowd（群衆）＋Sourcing（調達）の造語であり，「企業，組織が，自社もしくはアウトソースの人材により実施していた業務を，よりオープンかつ不特定多数のCrowd（群衆）から人材を集め実施すること」と定義されている．企業などが目的（需要）を提示し，それを不特定多数の情報発信者が参加して解決（供給）することで大量の作業を効率よく処理することが目的である（図1.1）．従来は不特定多数の人間に対して目的を提供，結果の収集を行うことが難しかったが，インターネットの技術革新に伴い可能となった．

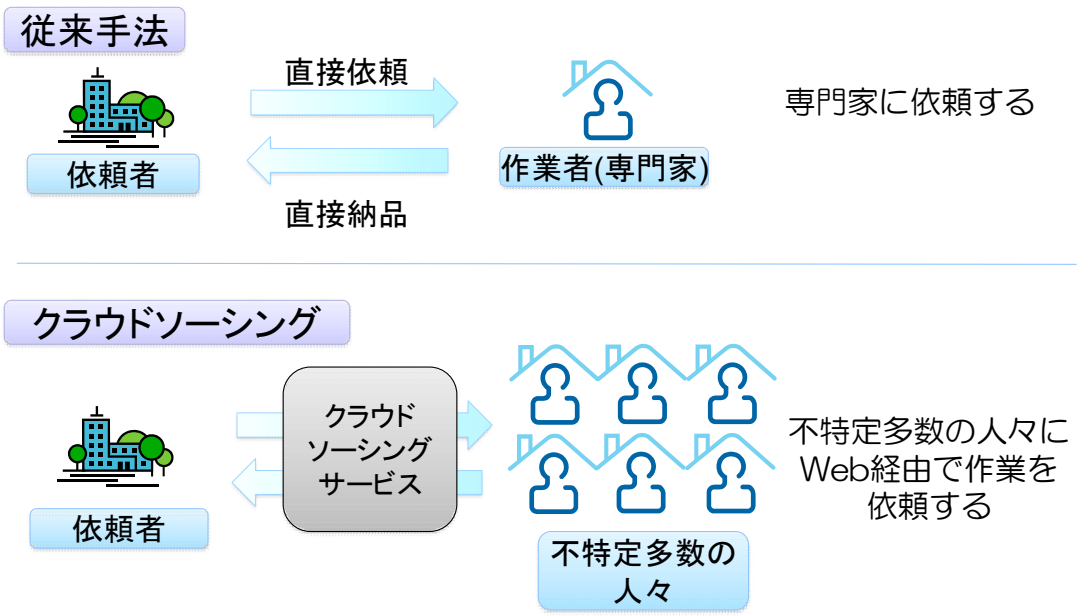


図 1.1: クラウドソーシング概要

大学・企業等の研究機関では，このクラウドソーシングの技術を様々な研究データの解析に用いている．研究データの作成は精度的な問題から自動化出来ないケースが多く，研究者，もしくは専門の技術を持った外部の業者といった人手による作業が必要になる．しかし，昨今の研究に用いられるデータはビッグデータと称される大量なデータであることが多い．従来の人手による作業では巨大データを扱うにはコスト，速度の面から難しくなってきている．そこで，我々はクラウドソーシングを用いている．

既存のクラウドソーシングサービスとして Amazon Mechanical Turk[1] や Yahoo!クラウドソーシング[2] などの様々なサービスが存在する．しかし，これらの外部サービスを研究データの作成に利用するにはデータの機密性の保持と精度の面から問題があった．企業が保持する研究データは秘匿性が高いデータが多く，外部のサービスに委託するには機密上，様々な点で問題が発生する．このような企業内のデータの機密性を保持するためには外部のサービス利用では難しい．さらに，我々は作業（タスク）の処理結果を研究データとして用いている．データを利用する目的は様々であり，自然言語処理における言語モデルの構築など，得られたデータから統計的なデータを用いるため精度の重要性が低いケースから，実験結果データの評価，自然言語処理における辞書データの構築など精度の重要性が高いケースまで様々なケースがある．このような精度の重要性が高いケースでは作業結果の品質を高く維持しなくてはならないが，そのためには外部のサービスが提供している機能の範囲では十分ではなく，さらに外部のサービスに新規の機能を追加することも難しい．我々はこれらの問題を解決するために，クラウドソーシングシステムを機密性が高くデータの安全性を高めることが可能なプライベートな環境下において構築することで問題の解決を試みている．

本研究では，フィルタリング手法を組み合わせることでコスト面を考慮しつつ，ワーカーを効率的にコントロールする精度向上手法を提案した．また，プライベート環境下において精度向上手法を適用した独自のクラウドソーシングプラットフォームを用いて，実際に実務に適用することで精度向上が可能であることを確認している．

また，利用範囲の拡大に従い実際にタスクを処理するワーカーの数も増大しており，将来的にクラウドソーシングにおける作業が社会における一つの就労形態となることが予想

[1] https://www.mturk.com/mturk/
[2] http://crowdsourcing.yahoo.co.jp/

される．しかし，そのような傾向にあるにもかかわらず，現状のクラウドソーシングではワーカーに対する育成や労働環境の改善と言ったサポートが十分であるとは言い難い．これはワーカーが不特定多数であり，補充や変更が容易であることが原因であると予想されるが，このようなワーカーの安易な変更は，ワーカーの経験不足による全体の精度低下やクラウドソーシングの市場の縮小という問題につながりかねない．

そのため今後のクラウドソーシング運用では通常の労働環境と同様に人材（ワーカー）の育成が重要になってくると予想される．しかし，クラウドソーシングにおける人材育成には様々な問題がある．特にマイクロタスク型クラウドソーシングではワーカーの数の多さ，ワーカーの匿名性からワーカー個人への対応が難しい．また,「高速」「低コスト」が利点であるため，コストや時間をかけて人材を育成するのはその利点を失う可能性がある．

我々はこのようなマイクロタスク型クラウドソーシングにおける人材育成の問題に対し，ワーカーがタスクを処理する過程で適切な学習タスクをこなすことで能力を向上させる段階的な学習方式を提案する．このような段階的な学習法式としては，学習支援システム（Intelligent tutoring system，ITS）における学習モデルをベイジアンネットワークによって表現する研究 [Ueno 00] が提案されており，その有効性が示されている．我々はこのベイジアンネットワークを用いた段階的学習手法のマイクロタスク型クラウドソーシングへの適用を提案する．具体的な手法としてはワーカーのタスク処理結果からベイジアンネットワークを用いてタスク間の関係性を解析し，タスクを処理することで段階的な学習が可能となるような学習タスクを自動生成することでワーカーの能力の育成を狙う．

このように低品質なワーカーを高品質なワーカーへと育成することで，安易なワーカーの排除を行うこと無く，精度向上と同時にワーカーの労働環境を向上させることが我々の狙いである．

本研究では，クラウドソーシングという手法に関する分類，学習やクラウドソーシングにおいて機械学習的なアプローチを行った既存の研究に関して紹介し（2 章），我々の提案する独自のマイクロタスク型クラウドソーシングに関して述べる（3 章）．そして，マイクロタスク型クラウドソーシングの精度向上手法として，ワーカーフィルタリングによる手法を提案し（4 章），ワーカーフィルタリングによる効果に関する考察を行う（5 章）．また，マイクロタスク型クラウドソーシングの精度向上手法として，クラウドソーシングに

おける段階的学習手法を提案し（6章），段階的学習手法に関するPCSS上の実験とその効果に関する考察を行う（7章）．そしてこのような精度向上手法を採用したマイクロタスク型クラウドソーシングを用いて行った自然言語処理のための語彙収集に関して紹介し（8章），最後にまとめと今後の課題に関して述べる（9章）．

第2章　関連研究

本章では，本研究で取り扱うクラウドソーシング関連の研究に関して述べる．まずは既存のクラウドソーシングに関する概要を説明する．そして，クラウドソーシング全体に関する研究に関して説明し，次にクラウドソーシングの精度向上手法に関する研究に関して説明し，さらにクラウドソーシングにおいてワーカーの学習や機械学習を用いた研究に関して説明する．

2.1　クラウドソーシングの分類

クラウドソーシングは適用することで大規模データの処理が可能になる，コストを低下させることができるなどの利点から大きな注目を浴びている分野である．そのためクラウドソーシングの名前を持つサービスや研究は数多くあるが，不特定多数の人間がひとつの目的に対して共同で作業を進めること全般をクラウドソーシングと指すこともあり，非常に幅広い定義となっている．その為クラウドソーシングの分類に関しても様々な分類方法があるが，本研究では作業の規模とかかるコストを軸に分類を行い，それぞれ (1) コンペティション型，(2) ジョブマッチ型，(3) マイクロタスク型，(4) ボランティア型と定義した．これらを図 2.1 のように位置づけている．

1. コンペティション型

   企業や組織が大規模，高難易度の作業を提示し，そのタスクに対して一人，もしくは複数の人間が解決を試みる．作業は非常に難易度が高く，明確な解決方法は作業を提供した企業や組織にもわかっていない．その為アイデア募集や研究目的に使われることが多く，作業を処理する作業者も専門家など高スキルを持っている人間が多い．また，作業の終了までにかかる時間も多大である．謝礼は問題を解決した作業者にのみ支払われ，それ以外の作業者には支払われないという点が特徴である．また，その

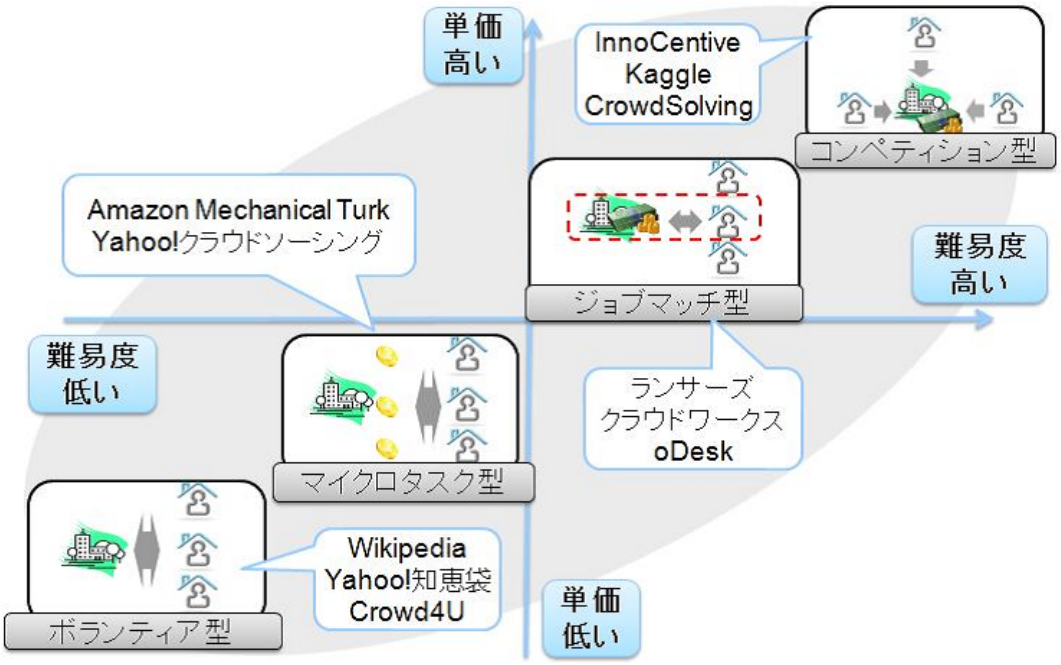


図 2.1: クラウドソーシングの分類

謝礼額は非常に高く賞金として扱われることが多い．企業が研究開発を外部に委託する InnoCentive[1]，データ解析，分析を外部に委託する Kaggle[2]，CrowdSolving[3] などがこの形式のクラウドソーシングを行っている．

2. ジョブマッチ型

企業や組織がコンペティション型程ではないがスキルが必要な比較的難易度の高い作業を提示し，その作業に対して複数の作業者候補が応募を行う．作業を提示した企業や組織は募集に応じた作業者候補から条件にマッチする作業者を選び契約を行うことで作業を進める．作業の獲得には作業者同士での競争は発生するが，契約がなされた以降は競争が発生せず作業を完了させれば謝礼が支払われるのが特徴である．この形式は Web デザインや文章作成など目的に応じて様々な企業が参加してお

[1]http://www.innocentive.com/
[2]http://www.kaggle.com/
[3]https://crowdsolving.jp/

り，ランサーズ[4]，クラウドワークス[5]，oDesk[6] など様々な数多くの企業がこの形式のクラウドソーシングを行っている．

3. マイクロタスク型
   企業や組織が用意した大量の難易度の低い作業を，数多くの不特定の作業者が作業を行う．作業の難易度は低く，一つの作業にかかる時間は数秒から数分と非常に短いが，支払われる単価も低く設定されており大量に作業を処理することが前提となっている．そのため作業者は特にスキルを必要としない場合が多い．大量の作業と大量の作業者を維持するコストが発生する．Amazon Mechanical Turk[7]，Yahoo!クラウドソーシング[8] がこの形式のクラウドソーシングを行っている．

4. ボランティア型
   不特定多数の人間に作業を提示するが，謝礼は発生せず作業者のボランティアによって行われるタイプのクラウドソーシングである．その為作業の内容など作業者のモチベーションを維持する方法が重要となってくる．Wikipedia[9] や Yahoo!知恵袋[10] や Crowd4U[11] などがこの形式のクラウドソーシングを行っている．

このようにクラウドソーシングは様々な企業組織が様々な目的で行っており，作業を出題する側，作業を処理する側ともに参加が非常に容易になっている．とくに米国では非常に大規模化しており市場全体で 10 億ドルを超えるともいわれており新しい雇用形態として非常に注目されている（図 2.2）．その反面，作業を行う人間が不特定であるため精度の維持が難しい，適正な賃金の設定が明確になっていないなど様々な問題もあり，クラウドソーシング自体が研究対象としても注目を集めている．

[4] http://www.lancers.jp/
[5] http://crowdworks.jp/
[6] https://www.odesk.com/
[7] https://www.mturk.com/mturk/
[8] http://crowdsourcing.yahoo.co.jp/
[9] http://ja.wikipedia.org/
[10] http://chiebukuro.yahoo.co.jp/
[11] http://crowd4u.org/

（単位：百万円）

	2011年度	2012年度	2013年度	2014年度	2015年度	2016年度	2017年度
クラウドソーシング市場規模	212	443	990	1,730	2,740	4,160	6,020
（前年度比）	161.8%	209.0%	223.5%	174.7%	158.4%	151.8%	144.7%
（CAGR）	–	109.0%	116.1%	101.3%	89.6%	81.4%	74.7%

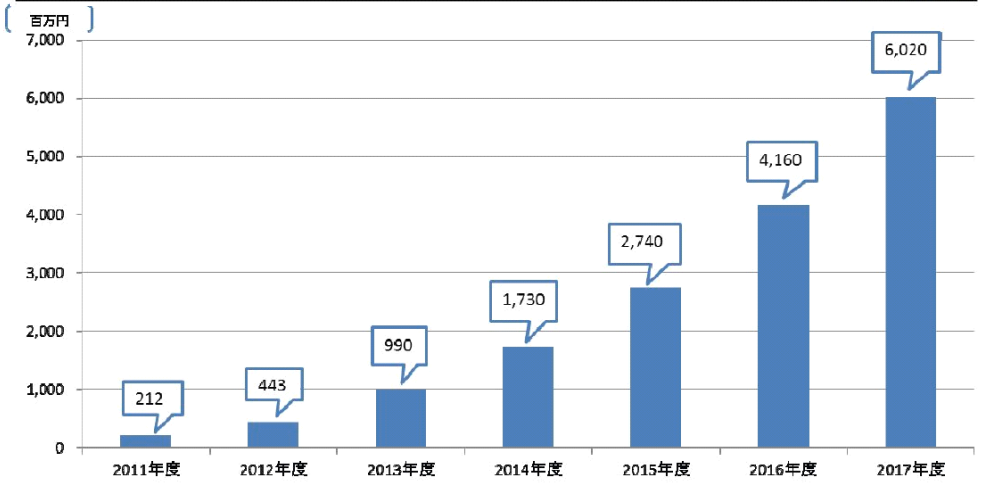


図 2.2: クラウドソーシング市場規模推移予測（2011～2017 年度） [Yano 13]

2.2　クラウドソーシングの利用に関する研究

クラウドソーシングはビジネスとしての目的だけではなく研究対象として，または研究のためのツールとして様々な分野から注目されている．近年でもデータマイニング系の Knowledge Discovery and Data Mining 2012 (KDD2012)，KDD2014，情報検索系の Special Interest Group on Information Retrieval 2010 (SIGIR2010)，SIGIR2011，SIGIR2014，SIGIR2016，画像処理系の Computer Vision and Pattern Recognition 2010 (CVPR2010)，CVPR2014，言語処理系の North American Chapter of the Association for Computational Linguistics 2010 (NAACL2010)，翻訳系の Association for Machine Translation in the Americas 2010 (AMTA2010)，音声処理系の International Speech Communication Association 2011 (InterSpeech2011)，InterSpeech2015，ヒューマンコンピュテーション系の Human Computation and Crowdsourcing 2011 (HCOMP2011)，HCOMP2012，HCOMP2013，HCOMP2014，HCOMP2015，HCOMP2016 など様々な学会でクラウドソーシングのワークショップやカンファレンスが開催されており，クラウドソーシングに関する論文数も増

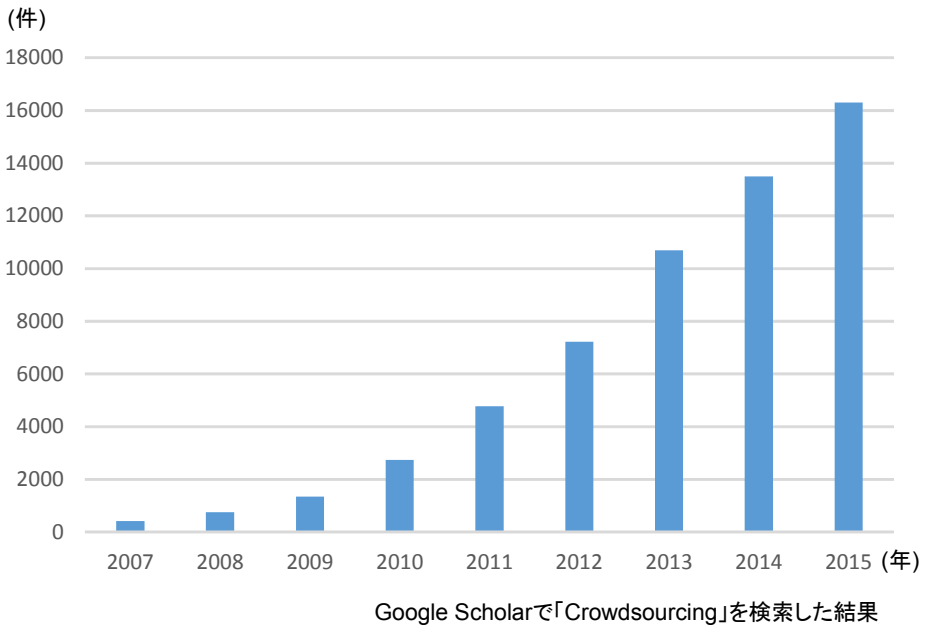


図 2.3: クラウドソーシング関連の論文数の推移

加している（図 2.3）．しかし，クラウドソーシングはその定義が広義であるため研究内容もまた多彩である．本章ではまずクラウドソーシングがどのように利用されているかを述べるために，クラウドソーシングをツールとして用いた研究に関して述べ，その後，本研究に関連性の高いクラウドソーシング自体の研究に関して述べる．

クラウドソーシングをツールとして用いた研究としては，得られたデータを他の研究に用いることで精度向上を図る研究，クラウドソーシングを教師データとして用いる機械学習に関する研究やデータ分析に関する研究が主に該当する．クラウドソーシングで得られる結果データはノイズを含むことが多いがクラウドソーシングのシステムを改善する以外の方法でノイズを除去し，学習アルゴリズムの改善に用いている．また，クラウドソーシングをツールとして用いた研究は数多く存在するため，クラウドソーシングによって得られた結果をどのように研究に利用するかという観点で目的別に以下の 3 つのカテゴリに分類した．

1. データ加工のための研究

翻訳，書き起こしなど元となるデータを他の形式のデータに加工することを目的と

した研究．元のデータの加工後のデータは意味的には同じである．元データに対して同価値である別形式のデータを付加するとみなして本研究ではアノテーション作業の一環として扱う．

2. データ詳細化のための研究
   画像，動画，テキストなど様々なデータに対して，人間が情報を付加することで情報量を増やすことを目的とした研究．付加したデータは元のデータと比較して規模は小さい．得られた付加情報は元のデータとともに学習データなど機械システムの精度向上に使われる．

3. データ評価のための研究
   研究によって得られたデータを直接人間が評価することで研究の精度を測ることを目的とした研究．付加されるデータは数段階の評価のみであり非常に小さい．直接人間の評価を得ることで精度を示す研究や，機械で自動化した処理の結果と人間が処理した結果を比較することで精度を示す．

よくある事例として，ヘルプデスクシステムの構築を例とした場合，以上のカテゴリ分類は以下のように対応付けることができる．

最初は顧客とオペレータの会話の音声データのみが存在するが，より扱いやすくするためにテキストデータへの書き起こしを行なう（データ加工のための研究）．得られたテキストデータに対してタグ付けやコメントを付け，管理，分類を行って FAQ に転用する（データ詳細化のための研究）． 結果として得られた FAQ がユーザにとって使いやすいものになっているかどうかの判定をユーザ自身に行わせる（データ評価のための研究）．

データ加工のための研究は翻訳，音声書き起こし，画像書き起こしなどが該当する．翻訳，書き起こしなどの作業は難易度が高く，また，一つ一つの作業が大きいことが多い．その為難易度が低く作業量が小さいマイクロタスク型のクラウドソーシングで直接処理した場合は結果精度が低くなってしまう．この精度を上げる方法が研究テーマとして取り上げられている．

代表的な手法として作業データを分割して粒度を下げることで精度を向上させる手法がある．この手法を翻訳に用いた研究 [Matthew 10]，また，同様に音声からの書き起こしに

用いた研究 [Evanini 10] がある．

また，マイナーな言語の翻訳をクラウドソーシングで行なうことによって処理速度を向上させる研究がある．多言語の語彙情報を収集する研究 [Ann 10, Audhkhasi 11]，韓国語，ヒンディ語，タミル語を対象とした研究 [Novotney 10]，スワヒリ語などを対象とした研究 [Gelas 11] などがある．

また，翻訳者同士でコミュニケーションを行い速度や精度を上げるシステムを導入して翻訳を行う研究 [Robert 10] などがある．

データ詳細化のための研究は画像，動画といったデータに情報を付与するためにはデータの意味を解釈する必要があるため，機械による自動処理では難しいことが多い．従来は機械に人間が持つ発想，解釈などの能力を模倣させ，大量のデータを処理させることを目的としていた研究が多かったが，クラウドソーシングの登場により直接大規模データを低コストで扱うことが可能となった．また，クラウドソーシングによる処理で得られた結果を機械処理にフィードバックすることで機械処理の精度向上を目指しているものも多い．直接解析，検索しにくい画像，動画に対して人間が関連するデータを付与し，付与されたデータをキーにして解析，検索することで精度を向上させている．また，動画，画像と比較して解析しやすいテキストに対しても要約や，意味解釈など自動では難しい処理に対してもクラウドソーシング作業が効果的である．

画像に対するアノテーションの研究としては，画像から連想される検索クエリを付与し画像検索精度を上げる研究 [Wanf 11]，画像に高精度な自由文書を付与する研究 [Cyrus 10]，画像に定められたデータセットでプロパティを付与して画像認識精度を上げる研究 [Ali 10] などが挙げられる．

動画に対するアノテーションの研究としては，動画にムードワードとレーティングを行って番組推薦の精度を上げる研究 [Mohammad 10]，動画に動作線を付与して画像認識の精度を上げる研究 [Ian 10] などが挙げられる．

音声に対するアノテーションの研究としては，音声へのタグ付けを行って音声検索の精度を上げる研究 [Luke 09] ，アクセントを付与する研究として，Facebook ゲームでアクセントを付与する研究 [Akasaka 09]，作業者のバックグラウンドを重視してアクセントを付与させる研究 [Kunath 10]，non-native の音声のアクセントを付与させる研究 [Evanini 10]，

音声を聞いてその音声の極性や感情を付与する研究 [John 12] ，作業者を選別してアクセント付与の精度を向上させる研究 [芦川 12, 芦川 13] などが挙げられる．

テキストに対するアノテーションの研究としては．ブログなどから内容のカテゴリを判定する研究 [Tae 10]，文章要約の精度チェックを行わせる研究 [Dan 10]．文章に感情を付与する研究 [Bart 10]，日本語の単語に対して読みなどの情報を付与し，自然言語処理の精度向上を行う研究 [芦川 12, 芦川 13] などが挙げられる．

データ評価のための研究はデータ評価のみではなくシステムの評価も行うためにクラウドソーシングを用いる研究である．デザイン，UI など明確な評価指標がないシステムにおいて結果の評価を行うには使用する人間が直接判定しなくてはならない．評価は評価人数が多いほど信頼度が上がるため，多様多数な人間が低コストで処理するクラウドソーシングが適している．正確に評価してもらうためにはクラウドソーシングの作業者は多様であるため，わかりやすく，かつシステム開発者の意図が正しく伝わるように作業を工夫しなくてはならずその点が研究対象となる．

対話システムをクラウドソーシングで評価する研究 [Yang 10, Jurcicek 11]，合成音声の結果をクラウドソーシングで評価する研究 [Wolters 10, Bucholz 11, Jeanne 13]，検索結果デザインをクラウドソーシングで評価する研究 [Dongqing 10] などがあげられる．

また，クラウドソーシングで行った作業結果に対して再度クラウドソーシングで評価することで，クラウドソーシング作業全体の精度を向上される手法も多い．この手法を翻訳に用いた研究 [Matthew 10]，また，同様に音声からの書き起こしに用いた研究 [Goto 11]，音声書き起こしから不良音声のフィルタリングまで行なう研究 [Lee 11] などがある．

これらの研究はクラウドソーシングをツールとして利用した研究であり，リクエスタやタスクの内容に大きく依存している．我々はタスクの内容に影響されることの無い，クラウドソーシングシステム全体の精度向上を目標としており，このような特定の用途に限定された研究では充分ではない．次節ではタスクの内容に依存しないクラウドソーシングの研究に関して述べる．

2.3 クラウドソーシングの精度向上に関する研究

マイクロタスク型のクラウドソーシングの性能を測る指標は数多くあるが，本研究では「コスト」「精度」「速度」をクラウドソーシングの性能を測る指標として考える．これらは相互に負の相関関係を持つことが多い．例えば，コストを下げるために報酬を下げるとワーカーのモチベーションに負の影響がでてタスクに対する処理速度が低下する．また，精度向上のために一つの問題を複数のワーカーに出題する場合において，コストを下げるためには一問あたりのワーカー数を減らさなければならず，結果として精度も低下するなどである．

マイクロタスク型のクラウドソーシングはその特性上,「安価で大量の処理が可能」という点に注目されることが多く，精度は優先度を低く設定されがちである．また，マイクロタスク型は一つ一つの作業の難易度が低いことも多く，精度を軽視させる要因の一つとなっている．しかし，クラウドソーシングの普及に伴い，タスクの内容が多様化し，精度に関しても高レベルの要求がなされつつある．

これまでもマイクロタスク型のクラウドソーシングの精度を向上させる方法に関して様々な研究がなされている．我々はこれらの研究を以下の 3 つのカテゴリに分類した．

1. タスクに対する精度向上手法
2. ワーカーに対する精度向上手法
3. 作業出題者（リクエスタ）に対する精度向上手法

(1) に関する研究はタスクのデザインに関する研究である．問題の表示方法や入力インターフェイスのデザインだけではなくタスクの進め方，出題方法などタスクに関する改善全般が該当する．タスクのデザインを改善することで精度向上につなげる研究 [Kittur 08]，タスクを複数に分割してワーカーの能力に応じて割り当てる研究 [松原 13]，タスクを複数のワーカーに出題し，結果を融合させることで精度を向上させる研究 [Dawid 79, Welinder 10, Whitehill 09, Mao 12]，ワーカーにタスク処理と同時に処理結果の精度への確信度を申告させる研究 [櫻井 12, 小山 13] などが行われている．既存のサービスにおいても正解が予め

わかっている問題をタスクに混ぜ，その結果を用いてワーカーの能力をはかり選別する手法 （Yahoo クラウドソーシング）[12] などが行われている．

(2) に関する研究は作業を行なうワーカーに関する研究である．ワーカーに信頼度の高いワーカーを紹介させる研究 [西 13]，作業結果を学習データとしてスパムワーカーを排除する研究 [Halpin 12]，ワーカーのタスクに非依存な行動からワーカーの能力を予測する研究 [Kilian 12]，ワーカーのランキングを行うことで低品質ワーカー，スパムワーカーを排除する研究 [Raykar 12]，データに対するラベリングを行なうタスクにおいて高品質ワーカーと低品質ワーカーを判定する閾値を算出することで，低品質なワーカー排除し最適なデータを得るためのを研究 [Donmez 09] などが行われている．既存のサービスにおいても，ワーカーに事前テストを受けさせてリクエスタが必要に応じてワーカーを選別する手法 （Amazon Mechanical Turk）[13] などが行われている．

(3) に関する研究はタスクを提供するリクエスタに関する研究である．不適切なタスクはワーカーのモチベーションを下げ，結果としてワーカーの品質低下につながる．この不正リクエスタを排除することで全体の精度を保つ研究 [馬場 13] などが行われている．

また，(1) と (2) の組み合わせである，事前に事前テストを受けさせてワーカーを選別し，さらに出題方法の調整でワーカーを選別する研究 [Kazai 11] なども行われている．しかし，この研究では特定のタスクを対象としたリクエスタ視点で行われている研究であり，複数の種類のタスクが発生した場合は対応が難しいという問題がある．

PCSS では主に (2) のワーカーに対する精度向上手法を中心に行っている．(1) に関してはシステム外の精度向上手法に関する事項であるため，タスク内容に依存することが多くシステム側で対応しにくいという問題がある．実際に PCSS を運用するにあたってはリクエスタのタスクの内容に応じて対策を行っているが，PCSS における機能とは異なるため本研究では触れない．また，(3) に関してはプライベートなクラウドソーシングという特性上リクエスタが明確であるため，不正なリクエスタは存在せず対策は不要である．

[12] http://crowdsourcing.yahoo.co.jp/
[13] https://www.mturk.com/mturk/

2.4 ICT を教育に用いた研究

ワーカーの精度を向上させる手法としてワーカーを生徒とみなして教育を行う手法が考えられる．しかし，クラウドソーシングにおけるワーカーは不特定多数であるため直接指導を行うことは現実的ではなく，インターネット経由で行うなど ICT を用いた教育が必要となる．ICT を用いた教育に関しては様々な研究が提案されてきた．ICT を用いた学習支援システム研究の一例として下記の分類が提案されている [川合 88]．

- ドリル＆プラクティス型
  生徒が既に学習した内容を復習したり，強化したりすることを目的としている．電子制御のドリル形式で教育を行う．解答の正誤によって出題の難易度を変化させるので，学習者のレベルに合った演習が可能である．イリノイ大学の PLATO（Programmed Logic for Automated Teaching Operations）[14] などがある．

- チュートリアル型
  いわゆる電子紙芝居（文章や図表や動画を統合したマルチメディア教材）．教科書的な知識を表示して学習させ，テストして結果を確認し必要に応じて再学習をさせる形式．多くの e-learning がこの手法を採用している．

- ゲーム型
  良い環境を与えればそこから知識を獲得するという考え方（構成主義的学習理論）をベースとし，受動的な学習だけではなく能動的な学習を促進するべくゲームを取り入れた学習法．

- シミュレーション型（マイクロワールド型）
  実際の画面を模倣し，得た知識を応用することで得た知識の深化を目的としている．ICT を用いた疑似実験環境を用いた学習法．

- 問題解決型
  与えられた課題に対してシステム側に指示を行い，得られた結果から判断してさら

[14] http://platohistory.org/

に指示を出すという作業を繰り返すことで問題解決を行う．その過程で問題解決に必要な様々な内容を学習することができるという学習法.

- ワードプロセシング
  コンピュータを使用して文章を作成する過程で，修正，文字変換などの言語操作を行うことで筆記，綴り字，句読法などを学習する学習法.

また，近年では教師が学習者に知識を伝達することを中心とした従来の学習観に対して，学習者が学習過程の中で知識の意味や価値に気づき，それらの知識を融合・統合させて新たな知識・概念・スキルを獲得することを中心とした学習観が中心となり，様々な学習者が共同作業を通して，知識を構築，取得していく学習を支援する.

- 協調学習型
  教師から学習者への教育だけではなく「活動の場」を提供し，コミュニティに参加することで学習者が相互的に知識を高め合う学習法.

協調学習型の学習方式を支援するシステムは様々なものがあり，CSCL（Computer Supported Collaborative Learning）と呼称される．CSCL に関する研究は数多く存在するが近年 AAAI や CSCW にて発表された研究として，高校生に解集合プログラミングを教えるためのオンライン学習環境に関する研究 [Reotutar 16]，ロールプレイングゲーム形式での AI 教育環境の提供と評価に関する研究 [Sintov 16]，AI 教育に必要な数式，図などを利用しやすくした学習支援環境 Moro に関する研究 [Singh 16]，通信教育における人工知能学習のためのカリキュラムデザインと実施に関する研究 [Goel 16]，プログラマ以外でもデータ解析ができるようにするワークフローベースのデータ解析方法学習支援システムに関する研究 [Gil 16]，動画に様々なアノテーションを用いて時系列や内容のポイントを分かりやすくした学習支援システム TrACE に関する研究 [Dorn 15]，学習支援環境 Peer 2 Peer University (P2PU) における効果的な講座の作り方に関する研究 [Ahn 15]，Github のコミュニケーション能力や協調作業に注目した学習ツールのとしての可能性検証に関する研究 [Zagalsky 15]，マルチモーダルなアノテーションを可能にした生徒と教師のコミュニケーション及び教育補助ツール RichReview++に関する研究 [Yoon 16] 等がある．また，大規模な公開オンラインコースである MOOCs（Massive Open Online Courses）に関する研究として，試験監

督フレームワーク，カンニングなどを防ぐ手法に関する研究 [Li 15]，MOOCS のインストラクター側の問題意識に関する研究 [Zheng 16]，MOOCS の実際のエンプロイアビリティ（企業が雇用候補者を雇用する際に雇用候補者が持っている雇用に値する能力）に対する効果に関する研究 [Dillahunt 16] などがある．また，クラウドソーシングシステムを学習補助環境として用い，タスクを学習項目として新たなスキルを学習させる手法に関する研究 [Glassman 16] などもある．

クラウドソーシングワーカーを学習者とみなした場合，このような協調学習支援環境を用いてクラウドソーシングワーカーの教育を行うことは有効であることが予想される．しかし，クラウドソーシングのタスク内容は多岐に渡り，数多くのリクエスタがタスクの作成を行っている．それら全てのタスクのテーマに沿った協調学習支援環境の開発をシステム管理者側で開発するのはコスト面で現実的ではない．また，リクエスタ側が協調学習支援環境を作成すると仮定した場合も，リクエスタに負荷をかけることはリクエスタがクラウドソーシングに期待するコストの低さ，速度の速さと言った利点を損なってしまうなどの問題が存在する．そのため，クラウドソーシングシステムに協調学習支援環境を用いるためには，リクエスタにもシステム管理者にも負荷が少ない手法のさらなる研究が必要と考える．

2.5 クラウドソーシングや教育に機械学習を用いた研究

前節で ICT を教育に用いた関連研究を紹介したが，近年教育に機械学習的な手法を用いた研究も注目されている．クラウドソーシングに限らず，教育に機械学習的な手法を用いた研究として，

1-1 様々な教育の要素が生徒にどのように影響するかを推測する研究

1-2 生徒の状態から要因を推定する研究

1-3 生徒を分類して最適な教育プランを検討する研究

などがある．

1-1) に関連する研究として，生徒に対して実施したテストや手法がどのような効果があるかを推測する研究 [Xenos 04]，生徒の学習スタイルが最終的にどのように成果に影響し

ているかを推測する研究 [Garcia 07]，複数の教育手法が生徒にどのような影響があるかを推測し，図示する研究 [Fernandez 11] がある．

1-2) に関連する研究として，生徒の状況から社会的経済指標を計算する研究 [May 06]，生徒の家庭環境や収入から生徒の生活背景がどのようなものかを推測する研究 [Hoogerheide 12] がある．

1-3) に関連する研究として生徒をスキル別にグループ分けする研究 [Pardos 10, Almond 09] がある．

我々の研究はワーカーを生徒とみなした場合，どのような要因が生徒の能力向上に影響するかを推測する研究であるため 1-2) のグループに属している．

また，クラウドソーシングに機械学習的な手法を用いた研究として，

2-1　ワーカーを処理結果を解析することで分類する研究

2-2　一つの作業を複数のワーカーに処理させる過程で結果のマージを行う研究

2-3　一つの作業を複数のワーカーに処理させて得られた結果をグループ分けする研究

2-4　得られた結果から出題タスクの難易度や品質を推測する研究

などがある．

2-1) に関連する研究として，ワーカーを精度に応じてグループ分けする研究 [Wauthier 11, Venanzi 15, Nushi 15, Shaw 11]，作業結果の精度に応じてワーカーの精度を判定し，排除するべきワーカーを判定する研究 [Wais 11]，作業結果の精度に応じてワーカーのスコアリングやランキング付けを行う研究 [Shaw 11, Raykar 14, Burnap 13]，作業結果の精度に応じてワーカーの最適な報酬を推測するための研究 [Xie 15] がある．

2-2) に関する研究として，一つのタスクに対して複数のワーカーから得られた結果からマージされた最適な答えを取得することを目的とした研究 [Carpenter 11, Tang 11, Sun 12, Kamar 12]，得られた文章やツイートにおける一致率を計算し，それに応じて結果をマージする研究 [Simpson 15]，SNS やテキストなどへのラベリングデータをマージする研究 [Simpson 15] がある．

2-3) に関する研究として，一つのタスクに対して複数のワーカーから得られた結果を複数のグループに分類する研究 [Bragg 14, Tang 11, Hutton 12] がある．

2-4) に関する研究として，回答したワーカーのスキル，正解率などからタスクの難易度をモデル化する研究 [Bachrach 12]，ワーカーの正解率とワーカーのエラーレートからタスクの難易度をモデル化する研究 [Lin 12] などがある．

このようにワーカーの分類や結果の解析でクラウドソーシングの精度を向上させる研究は行われているが，我々のようにワーカーの行動履歴をベイジアンネットワークなどの機械学習的なアプローチで解析することでワーカーの品質を向上させる研究は行われていない．これは低品質なワーカーは排除することが一般的であることが原因であると考えられる．しかし，前述のように将来的にクラウドソーシングが就労形態として一般的になることを考えた場合，安易な排除は問題になることが予想される．そのため，ワーカーの学習に基づく精度改善による労働環境改善は重要である．

第3章　プライベートクラウドソーシングシステムの構築

本章では，本研究で開発を行った独自のマイクロタスク型クラウドソーシングに関して述べる．まずはクラウドソーシングにおけるタスクに関して説明し，システムの構築，及び既存のサービスとの比較に関して説明する．

3.1　精度向上手法の組み込みが可能なクラウドソーシングシステム

研究データの構築には大量のタスクを高速に処理しなければならず，そのために，我々は前章で述べたマイクロタスク型のクラウドソーシングを用いている．しかし，既存のマイクロタスク型のクラウドソーシングサービスを研究データ構築に利用するには精度の点に問題がある．

我々はタスクの処理結果を研究データとして用いるため作業結果の品質を高く維持しなくてはならないという点があり，そのためには外部のサービスが提供している精度向上のための機能の範囲では十分ではないことが多い．また，外部のサービスに精度向上のための新規機能を追加することも難しいという問題がある．

そこで，これらの問題を解決するために，プライベートな環境下において様々な精度向上手法を適用したマイクロタスク型のクラウドソーシングシステムを構築した．我々はこのクラウドソーシングシステムをプライベートクラウドソーシングシステム (Private Crowdsourcing System，PCSS) と呼称している．本章では PCSS の構築方法に関して述べる．

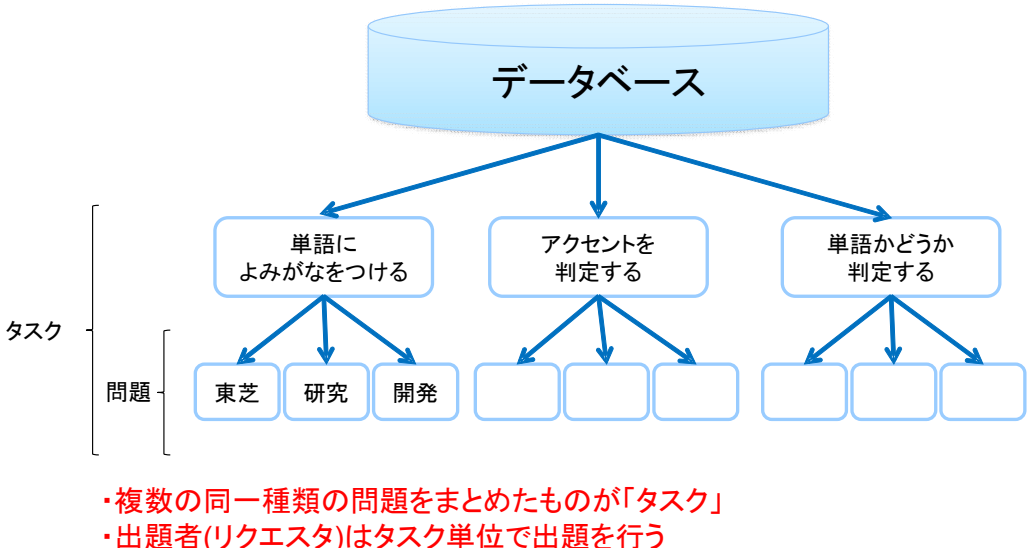


図 3.1: PCSS におけるタスク

3.2　PCSS 上で処理される作業の分類

クラウドソーシング上で処理する様々な作業はタスクと呼称され，様々なタスクが存在する．規模も研究テーマの考案のような大きなタスクから，アンケートなどの小さなタスクまで多岐に渡る．本研究ではマイクロタスク型のクラウドソーシングを対象としているため，処理が数秒から数分で完了するような小規模な作業が主な対象となる．しかし，作業のサイズが小さくなると個々の作業を管理するのは煩雑になるため，クラウドソーシングでは同様の小さな作業をまとめて処理することが多い．PCSS ではこのまとまりを「タスク」と呼称している．例えば図 3.1 における「単語に読み仮名をつける」作業を PCSS で行う場合，一つ一つの単語に読み仮名をつける作業を「問題」,「1000 問の単語に読み仮名をつける」という作業の集合がタスクとなる．リクエスタはこのタスク単位で PCSS に作業を出題する．

PCSS の主な利用用途としては研究データの作成であることは述べたが，大きく分けてデータの作成には何もないところからテーマやルールに従ってデータを作成する「データ収集・作成」系と既に存在するデータをベースに精錬化，別系統のデータへの変更などを行う「データ加工」系が存在する（図 3.2）．また,「データ収集・作成」系で作成したデータをさらに「データ加工」系のタスクで処理するケースも存在する．これらのタスクに関

図 3.2: PCSS におけるタスク例

する情報は PCSS 内のリクエスタ間で共有されており，既存のタスク作成やタスクシナリオにおけるノウハウを共有することで経験が少ないリクエスタも初回から精度の高い結果を得ることが出来ている．

3.3 PCSS の詳細と運用

本節ではプライベート環境下におけるクラウドソーシングの構築方法に関して述べる．クラウドソーシングは不特定多数の人によって動作するシステムであり，システムを構築しただけでは動作しない．システムに対してタスクを提供するリクエスタと，タスクを処理するワーカーが必要となる．システムは両者の仲介を行い，様々な面でサポートを行うことで全体的な効率の向上を図っている．

プライベートなクラウドソーシングを構築するにあたって一番の問題はワーカーの募集である．Amazon Mechanical Turk のように既に周知のサービスであればワーカーの募集は容易だが，無名の状態から必要な人数を集めるには多大なコストがかかる．一方，Amazon Mechanical Turk のように誰でも作業ができる環境ではワーカーの質を管理するコストが大きく，タスク結果の質が低下してしまうという問題もある．PCSS では，ワーカーの募集をネットワークリサーチを行なっているポイント業者へと委託した．ポイント業者は既にリサーチ対象となるユーザを数百万規模で管理しており，これらのユーザを PCSS のワー

カー候補とした．それらのワーカー候補に対して「作業可能な時間」「熱意」「希望時給」「学歴」「基本的な IT スキル」などのアンケートを実施し，各項目の能力が高いワーカー候補に対してプライベートクラウドソーシングへの案内を送付した．対象となったワーカー候補者の合計は 8 万人であり，これは PCSS におけるタスクの処理量が増えるに応じて募集を数回にわたって行った結果である．我々はこの絞り込みを「事前フィルタリング」と呼称している．これにより我々はポイント業者のユーザをワーカーとして作業を提供し，Web 経由で作業可能とし，さらにポイント業者を経由してワーカーに報酬を支払うという図 3.3 の構成を構築している．

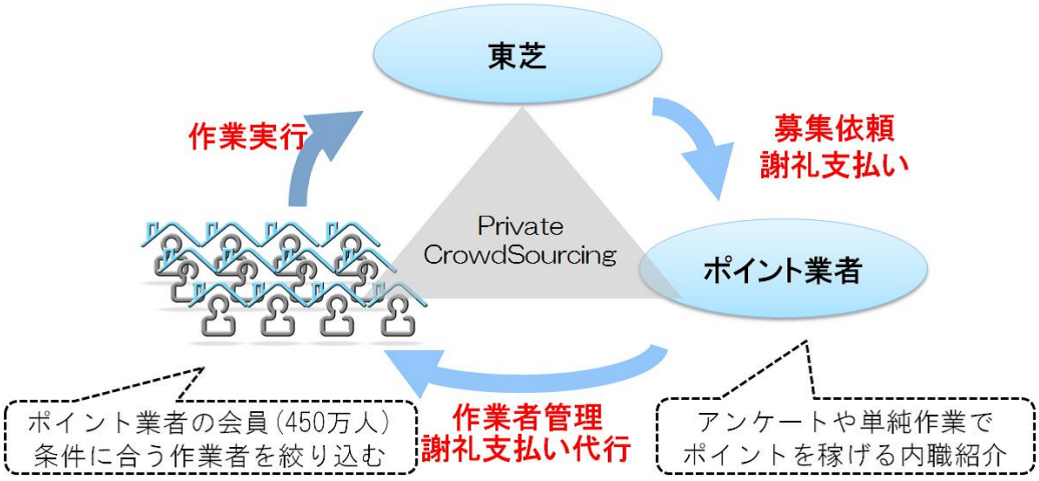


図 3.3: ポイント業者を経由したクラウドソーシング

システムは Perl で構築された CGI と，MySQL を用いたデータベースのサーバから構成されており図 3.4 のような構成となっている．リクエスタは Web インターフェイス経由でタスクをデータベースに登録し，ワーカーはデータベースに登録されたタスクに対して Web インターフェイス経由でタスク処理を行い，結果をデータベースに登録する．リクエスタはタスク処理が完了次第，結果をデータベースから取得する．次節ではこれらの流れをワーカー視点，リクエスタ視点で説明する．

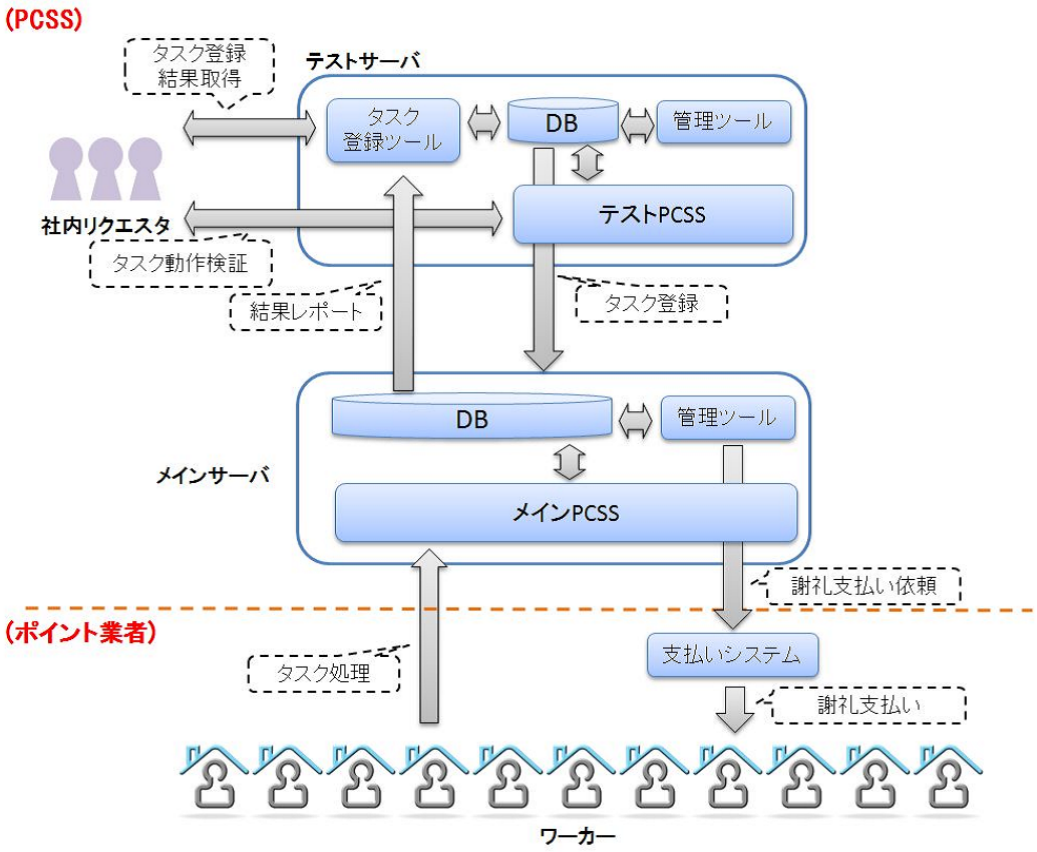


図 3.4: PCSS のシステム構成

ワーカー視点での PCSS における処理の流れ

図 3.4 の構成はワーカー視点では図 3.5 のようになる．ポイント業者が保有するユーザから抽出された新規ワーカーは PCSS のシステムにログインするための案内メールを受け取り，ログインを行う．ワーカーはログイン後以下のステップにしたがって作業を進めていく．

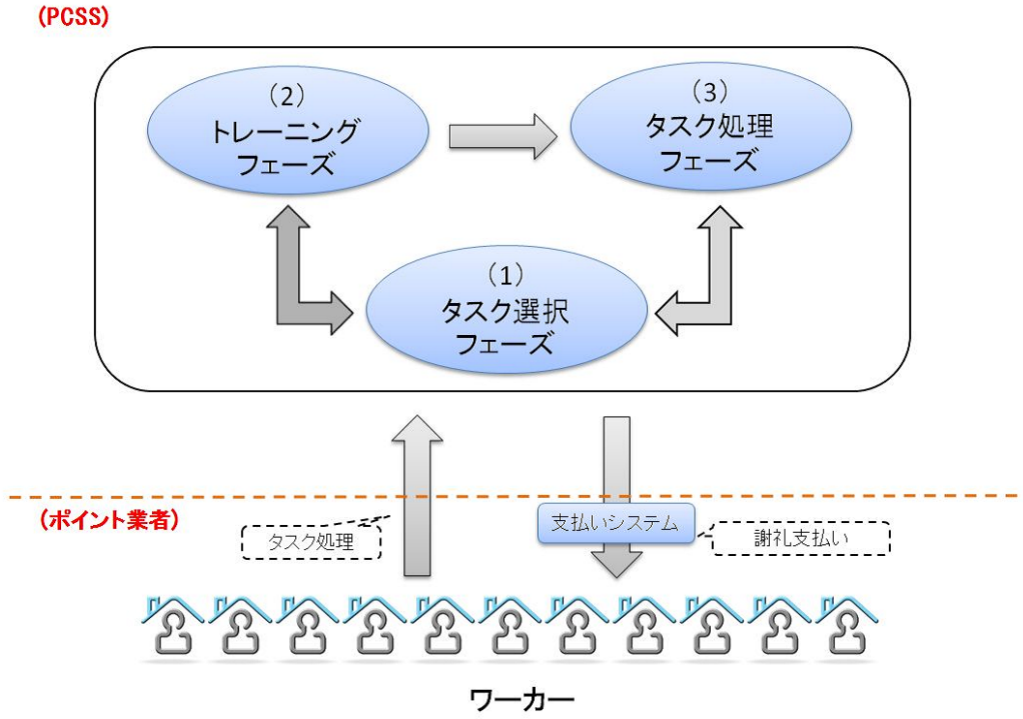


図 3.5: ワーカー視点での PCSS における処理の流れ

1. タスク選択フェーズ

   タスク選択フェーズは図 3.6 の左部における概要と賃金が併記されたタスクリストと図 3.6 の右部におけるワーカーの現時点における報酬額や正解率などを表示するステータス表示部からなる．ワーカーはタスクリストから概要を読んで作業したいと思うタスクを選択して作業を進めていく．ここで表示されるタスクの種類や順番はワーカーの特性や状態に応じてワーカーごとに変化するため，ワーカーによって処理可能なタスクは異なる．

図 3.6: タスク選択フェーズ

2. トレーニングフェーズ

ワーカーは各タスクにおいて，初回の処理を行う前にはトレーニングとして説明画面でタスクに関する説明を確認する必要がある．ここではタスクの概要や注意点などリクエスタがワーカーに注意して欲しいことを表示し，正しく作業ができるかどうか簡易なチェックを行うことができる．トレーニングフェーズの画面例を図 3.7 に示す．ワーカーはトレーニングの終了後 (3) タスク処理フェーズへ移る．また，トレーニングは後から繰り返し行うことも可能である．

図 3.7: トレーニングフェーズ

3. タスク処理フェーズ

タスクはリクエスタがデザインしているため，様々な入力インターフェイスが存在する．ワーカーは自分のステータスを確認しながら作業を進めていく．タスク処理フェーズの画面例を図 3.8 に示す．作業者は画面左の作業説明画面と結果入力画面に対して作業を進めていく．画面右には作業者のステータスが常時表示されており，作業者は自分のステータスを確認しながら作業を進めていくことができる．精度が一定以下になると作業ができなくなることを作業者に通知しているため，作業者は自分の結果精度が下がらないよう気をつけて作業を進め，結果として精度向上へつなげることができる．

図 3.8: タスク処理フェーズ

リクエスタ視点での PCSS における GUI 処理

図 3.4 の構成はリクエスタ視点では図 3.9 のようになる．リクエスタが PCSS を利用するにあたってはワーカーに処理してもらうタスクを作成しなければならない．PCSS ではタスクの作成作業を簡易化させるためにタスク登録ツール（図 3.10）をリクエスタに提供している．タスク登録ツール上では従来のタスクの情報を全て参照することができ，また，デザインをそのまま流用することも可能である．リクエスタは従来のタスクを参照し，以下のデータを作成する．

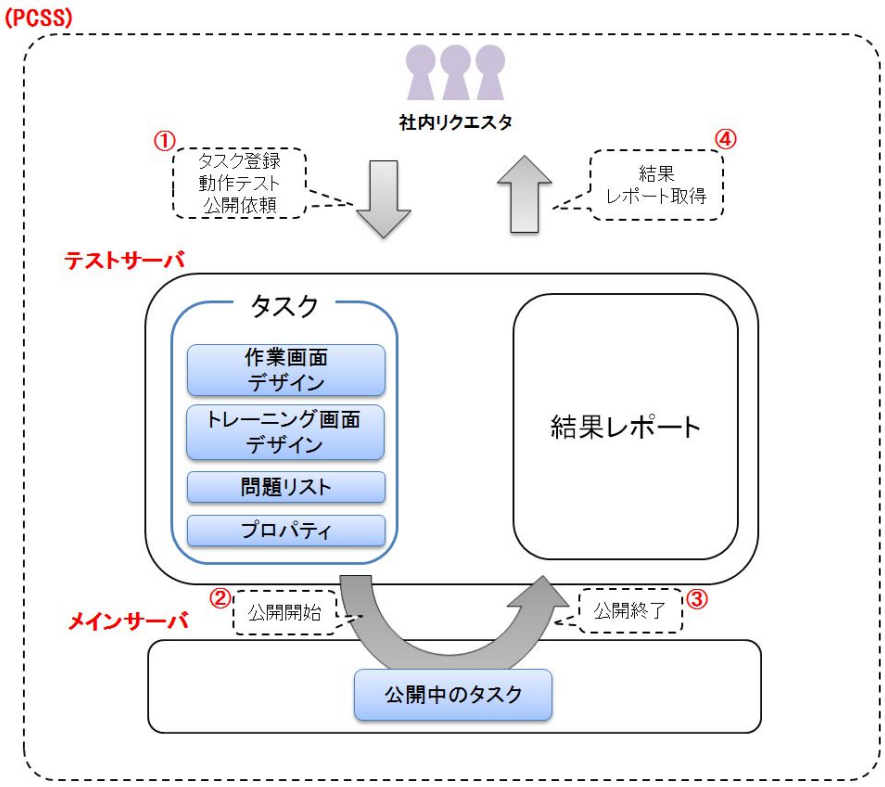


図 3.9: リクエスタ視点での PCSS における処理

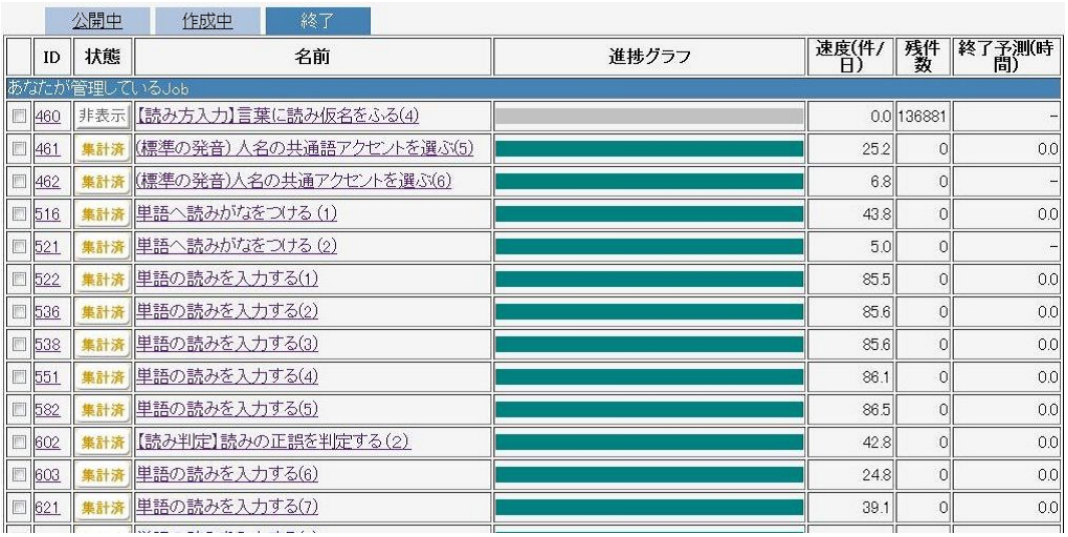


図 3.10: タスク登録ツール

1. 作業画面デザイン
   前節におけるワーカーの (3) タスク処理フェーズで利用する画面のデザインであり，html 形式で記述することができる．変数を利用することでタスクに関する可変な値を埋め込むことが出来る．

2. トレーニング画面デザイン
   前節におけるワーカーの (2) トレーニングフェーズで利用する画面のデザインであり，html 形式で記述することができる．

3. 問題リスト
   (2) で作成したタスクの作業画面デザインの変数部分に埋め込むタスクの値を列挙する．CSV 形式で記述することができる．

4. プロパティ
   タイトル，概要，ワーカーに提示する条件，必要な経験値，謝礼，ワーカーの能力（スキル），制限時間，判定方法（多数決または全員正解），予算を設定する．これらは過去のタスクを参照することでリクエスタが適正値を判断する．これらの値はタスクの処理中であっても速度向上，精度向上などの目的で適宜変更することが可能である．

リクエスタはこれらをタスク登録に必要なデータの作成後，図 3.9 における以下のステップに従って PCSS の利用を行う．

(1) 登録ツール上でタスク登録に必要なデータを登録，テストサーバにて動作確認を行いシステム管理者へ公開依頼を行う．

(2) システム管理者は公開依頼を受けたタスクの動作確認を再度行い，問題がなければメインサーバへ登録して公開を開始する．

(3) メインサーバでタスクの処理が完了すると，自動的にリクエスタの登録ツールへ終了ステータスが通知される．

(4) タスクのステータスが終了になったことを確認して，リクエスタは結果レポートの取得を行う．

リクエスタ視点での PCSS における WebAPI 処理

3.3 節で説明したようにリクエスタは用意された GUI を用いてタスクの出題から結果の取得までコントロールすることが可能である．しかし，GUI は自動化や他のアプリケーション，プログラムからの利用には向いていない．PCSS は他のサービスや製品組み込まれることも想定されているため，WebAPI を利用することで対応している．PCSS と API の関係性は図 3.11 のようになる．

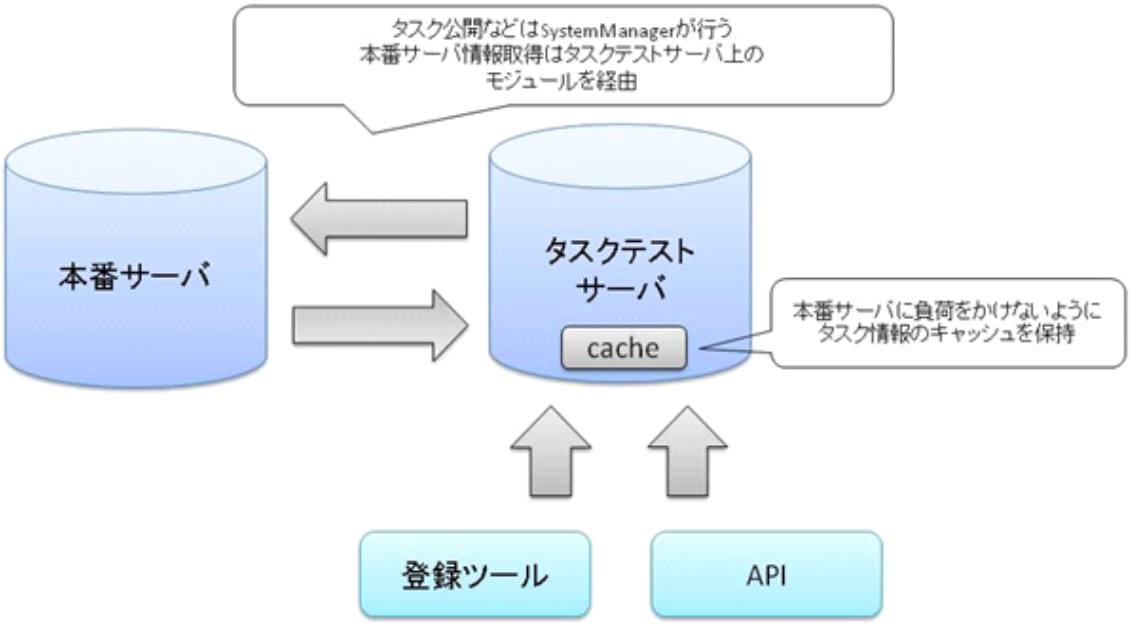


図 3.11: クラウドソーシング API の位置付け

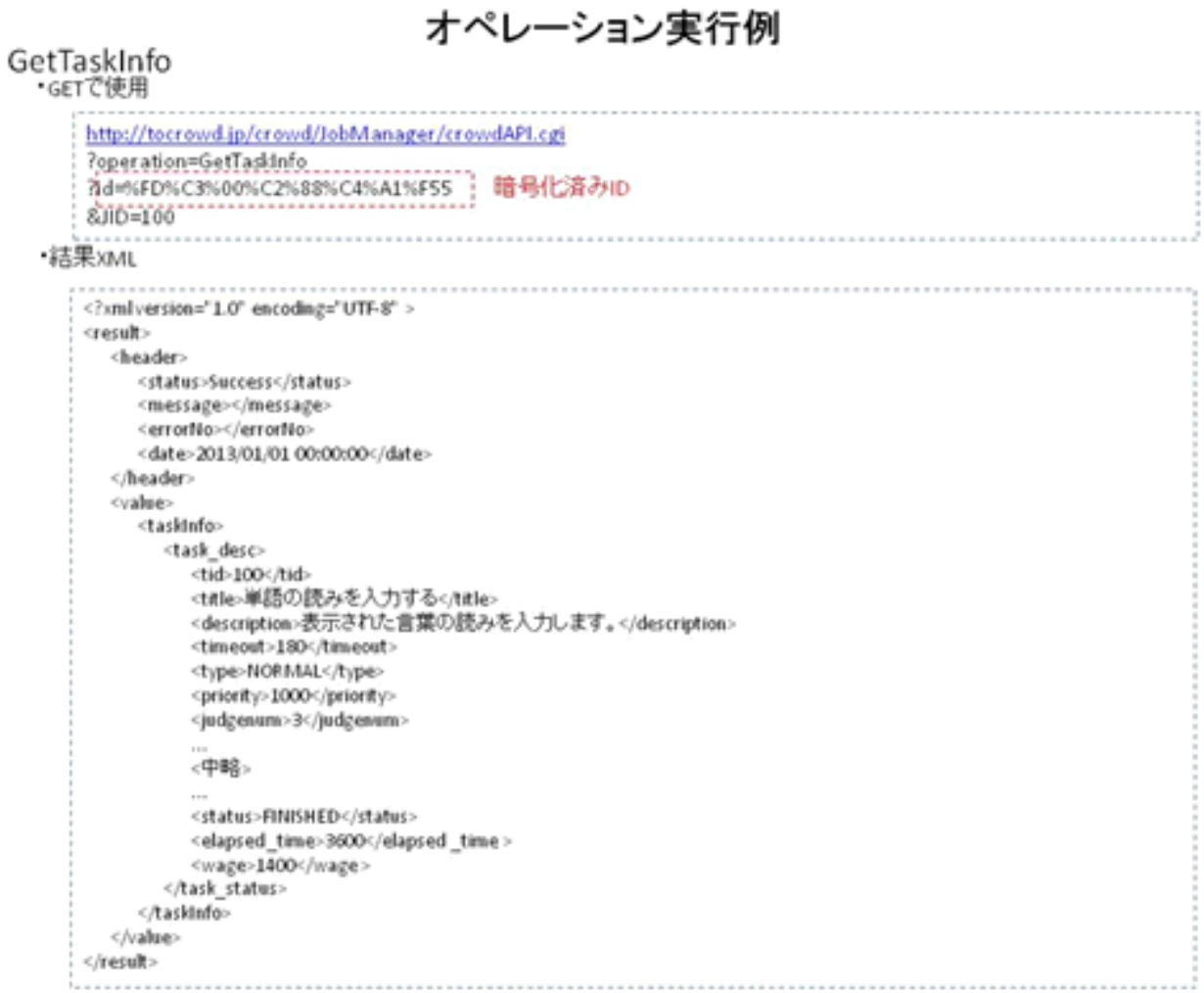


図 3.12: クラウドソーシング API の例

クラウドソーシング API は REST 形式で構築されている．REST とは，パラメタを指定して特定の URL に HTTP でアクセスすると，XML で記述されたメッセージが送られてくるようなシステムおよび呼び出しインターフェース（「RESTful API」と呼ばれる）のことを指す．システムの状態やセッションに依存せず，同じ URL やパラメタの組み合わせからは常に同じ結果が返されることが期待される形式である．クラウドソーシング API では処理用の CGI を用意し，CGI に対して必要なパラメタを渡すことで XML 形式の結果を得ることができる．例えば図 3.12 はタスクの情報を取得する処理である．クラウドソーシングではこれらの処理をオペレーションと呼んでおり，表 3.1 に示すオペレーションを扱う

ことができる．

表 3.1: クラウドソーシング API オペレーション

大カテゴリ	少カテゴリ	オペレーション名	内容
タスク系	作成、更新	CreateTask	タスク作成
		UpdateTaskDesing	タスクデザイン更新
		UpdateTrainDesign	トレーニングデザイン更新
		UpdateTrainedDesign	トレーニング終了デザイン更新
		UpdateTaskInfo	タスクデータ更新
		UpdateBinData	バイナリデータ更新
		CloseTask	タスク強制終了
		DeleteTask	タスク削除
		OpenTask	タスク公開
		ReuseTask	タスク再利用
	情報取得	GetTaskInfo	タスクステータス取得
		GetOpenJIDList	公開中のJIDリスト取得
		GetUnderConstJIDList	作成中のJIDリスト取得
		GetClosedJIDList	終了したJIDリスト取得
		GetBudget	予算上限取得
		GetUseAmount	現在の利用金額取得
		GetRequester	同じ予算グループのリクエスタ取得
		GetWage	時給取得
結果データ	情報取得	GetReport	結果データ取得
ワーカー	情報取得	GetWorkerInfo	ワーカー情報取得
テストデータ系	更新	DeleteTestResult	タスクテスト結果削除
	情報取得	GetTestResult	タスクテスト結果取得

以上のように外部のサービスを利用することなく独自の環境下においてタスクを出題しワーカーに作業をしてもらうプライベートなクラウドソーシング環境を構築することが出来た．本システムは 2011 年 11 月から運用を継続しており，表 3.2 に示す運用実績を持っている．

表 3.2: PCSS の運用実績

運用開始	2011 年 11 月
ワーカー総数	2454 人
毎月実績のあるアクティブなワーカー	150 人
問題数	1853 万件

3.4 PCSS 上で作業するワーカーの特徴

PCSS 上で作業するワーカーは 3.3 節で述べたように外部のポイント業者に募集を依頼している．ポイント業者は会員の個人名や報酬振込先や年齢，住所などの基本的な個人情報は有している．ワーカーの基本的な属性として，ワーカーの男女比（図 3.13），ワーカーの年齢分布（図 3.14），ワーカーの居住地比率（図 3.15）は図のようになる．ワーカーの最低年齢は 18 歳（18 歳未満は対象外としている），最高齢は 92 歳であるが，主な年齢層は 30 代から 50 代である．また，男女比はほぼ半々であり偏りはない．また，居住地は関東がほぼ半分を占めている．また，現在（2016 年 10 月時点）で正解数の多いワーカーのランキングを表 3.3 に示す．この表から高ランクワーカーは 30 代以上の女性の割合が多いことがわかるが，多様性を維持するためにも，年齢，性別，居住地では事前フィルタリングで絞り込むことは無く，その他の属性をベースにフィルタリングしている．

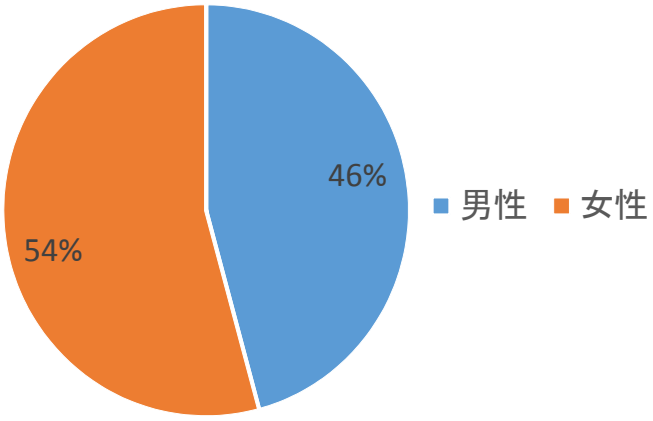


図 3.13: ワーカーの男女比

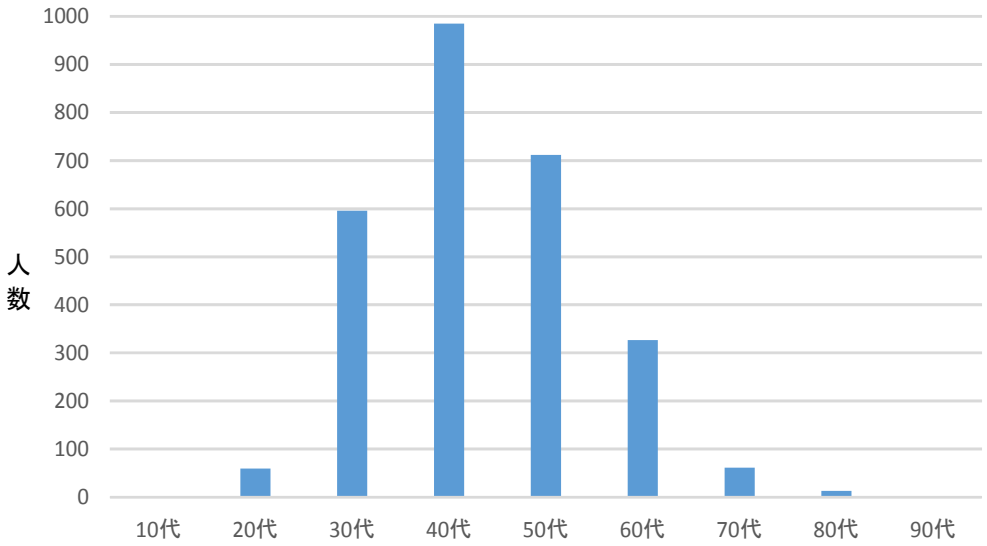


図 3.14: ワーカーの年齢分布

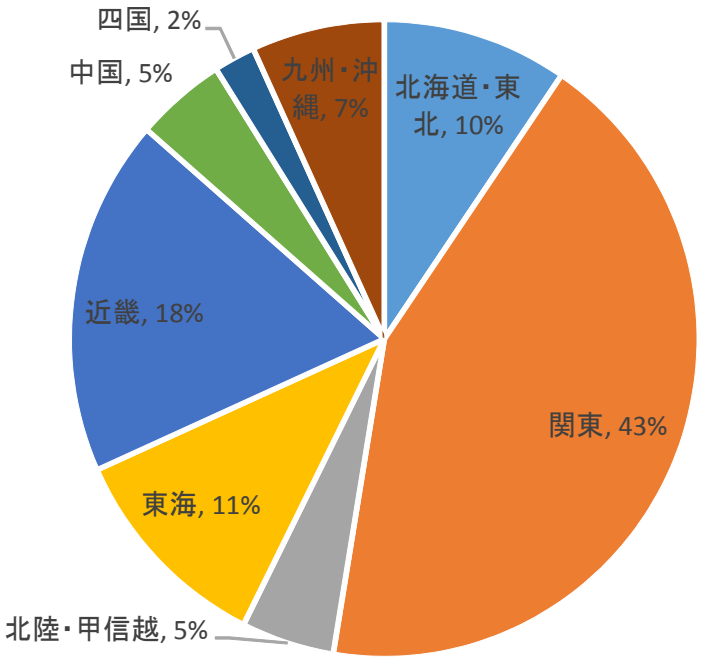


図 3.15: ワーカーの居住地比率

表 3.3: 高ランクワーカー

	正解数	性別	年齢
1 位	762742	女性	41
2 位	582100	男性	62
3 位	531695	女性	56
4 位	505217	男性	72
5 位	501370	女性	57
6 位	500154	女性	67
7 位	479527	女性	34
8 位	471750	女性	60
9 位	429687	女性	29
10 位	403276	女性	35

3.5 PCSS と既存のサービスとの比較

マイクロタスク型のクラウドソーシングでは最も有名なサービスとして「Amazon Mechanical Turk」が存在する．比較のために正解が存在している同一の作業を「本研究におけるクラウドソーシング（PCSS）」「Amazon Mechanical Turk」「作業員（一人）による手作業」で実施した．ここでの作業員は実際に眼前で作業を実施してもらっており不特定の人物ではない．比較に用いた作業内容は図 3.16 のように「画像を見て（性別，人種）を判定する」作業を 1000 問行った.「本研究におけるクラウドソーシング（PCSS）」「Amazon Mechanical Turk」では一つの問題を 3 人へと提供し，2 人以上が同一の解答を出した場合その回答を結果データとして扱う.「作業員（一人）による手作業」に関しては作業員が回答した解をそのまま結果データとして扱う．それぞれの環境で得られた結果データを正解データと比較し，精度比較を行ったところ結果は表 3.4 のようになった．結果として,「本研究におけるクラウドソーシング（PCSS）」は「Amazon Mechanical Turk」「特定の作業員（一人）による手作業」で作業したよりも精度は高い結果となった.「Amazon Mechanical Turk」よりも精度が高い理由としては「Amazon Mechanical Turk」が完全に不特定多数の作業者に対して作業を提供しているのに対し,「本研究におけるクラウドソーシング（PCSS）」は作業者を募集する際にフィルタリングを実施して作業員のレベルをある程度に保ってい

るためと推測される．また,「作業員（一人）による手作業」よりも精度が高い理由としては，一人で作業を行うよりも複数人で作業を行うことによって知識量が増えるためと推測される．

図 3.16: 比較のための人物画像判定タスク

表 3.4: Amazon Mechanical Turk と PCSS との精度比較

	PCSS	Amazon	特定作業者
性別判断	98.8%	93.3%	98.9%
アジア人判断	96.0%	80.0%	90.0%
白人判断	91.0%	69.0%	87.0%
黒人判断	99.0%	89.0%	97.0%

第4章　ワーカーのフィルタリングによる精度向上手法の提案

PCSS の環境を構築しただけでは得られるタスクの処理結果の精度は低いため，研究データとして使用するには十分ではない．本章では PCSS におけるワーカーフィルタリングによるタスク処理結果の精度向上手法に関して述べる．

4.1　ワーカーを対象とした精度向上手法

PCSS における精度向上手法は主にワーカーに対する管理を中心に行っている．クラウドソーシングは「不特定多数の外部の人間」に作業を委託する仕組みであるため，ワーカーの品質は様々である．特定のカテゴリのタスクにおける正解率が高い高品質ワーカーや正解率が低い低品質ワーカー，リクエスタの出題意図に沿った回答ができるスキル保持ワーカーや意図に反した回答をする負スキル保持ワーカー，全体の正解率が低いまたはスクリプトなどで処理を行う，システムから排除対象となるスパムワーカーなどのワーカーが存在する．既存のクラウドソーシングサービスでは数多くのリクエスタから数多くのタスクを受け入れているため，ワーカーが行うタスクは多種多様となり，結果としてタスク単位におけるワーカーの行動情報が少なくなり，ワーカーのコントロールが難しくなっている．PCSS ではプライベートという特徴上タスクのカテゴリが限られているため，タスクカテゴリに対するワーカーの行動情報は相対的に多くなっており，そのワーカーの行動情報を活かすことでワーカーの特性に応じた適切なタスクを与え，低品質ワーカーおよびスパムワーカーを排除することを可能としている．

以下にワーカーに対する PCSS の精度向上手法を (1) 事前フィルタリング，(2) 動的フィルタリング，(3) 結果フィルタリング，(4) 推測フィルタリング，の 4 つのカテゴリに分類した．それぞれのフィルタリングではコストと精度が異なり，コストが高いフィルタリング

は低品質ワーカーを排除する精度が高い．我々はコストの問題から，対象のワーカーの数に応じてフィルタリングを適用している．それぞれの手法はPCSSの運用における図4.1に示したタイミングで行われる．それぞれのフィルタリングに関して詳細を述べる．

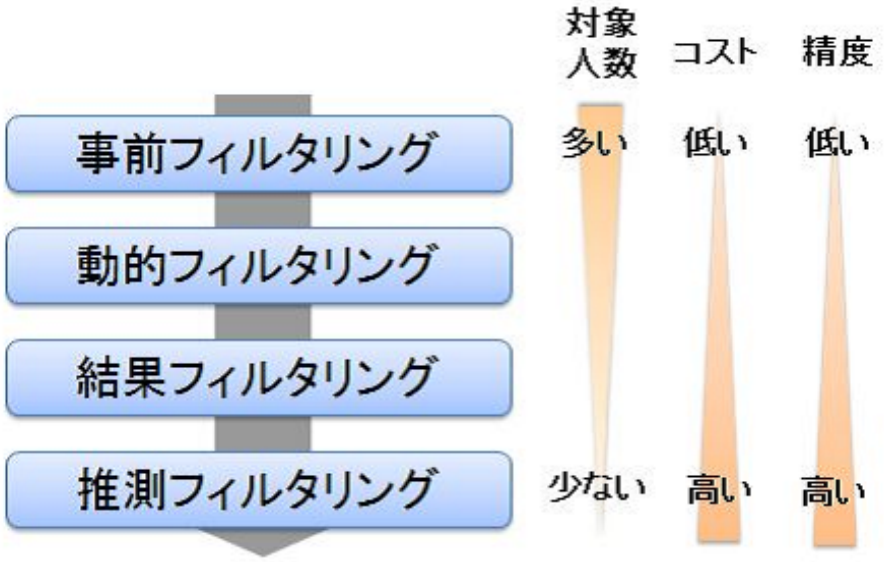


図 4.1: PCSSにおけるワーカーに対する精度向上手法

4.2 事前フィルタリング

ポイント業者からワーカーを募集する際に行うフィルタリングである．ポイント業者は数百万人の会員を有しており，これらのすべての会員をワーカーとして扱うのはコスト的に現実的ではなく処理能力的にも過剰である．また，これらの会員にはICTの素養が低い，Webにおける継続的な作業を望んでいない，などのPCSSに不適である会員も多く存在しており，このような明らかに高品質なワーカーになりえないワーカー候補を排除するために事前のアンケートを用いてフィルタリングを実施している（表4.1）．これらの質問により，実際にクラウドソーシング作業を行うことのできる時間（Q1，Q2，Q4），ベースとなるモチベーションの度合い（Q3），基礎学力（Q5），タスクに応じたスキルの予測（Q6），基本的なITスキル（Q7，Q8）などを測定することができる．また，特に「Q6:学生時代

に最も得意だった科目はなんですか？」の問では，音声処理系のタスクを中心に処理させたいワーカーの場合は「音楽」を，自然言語処理系のタスクをさせたい場合は「国語」を，POI などの地理情報を処理させたい場合は「社会」を選択したワーカーを優先させるなど，状況やリクエスタのニーズに応じて採用基準を変更している．また，アクセント付けなどの難易度の高いタスクでは，対象となるタスクを処理できるワーカーに特化して募集するために表 4.1 の問い合わせ内容に追加して，対象となるタスクの出題内容を一部出題することでフィルタリングを行う場合もある（図 4.2）（図 4.3）．

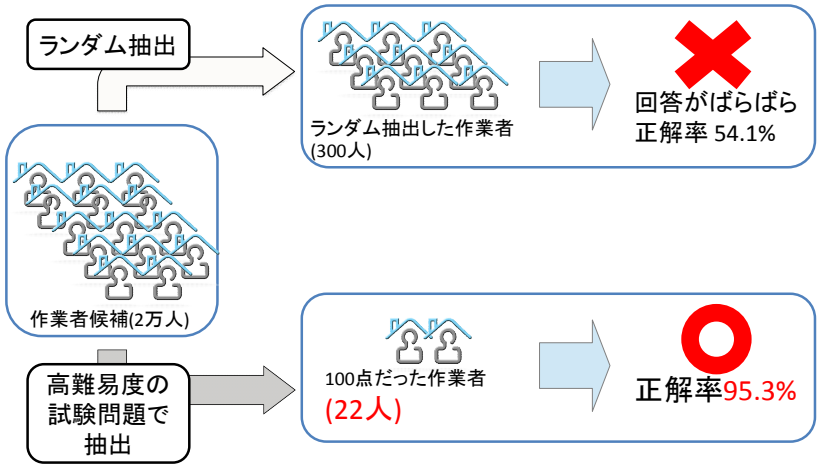


図 4.2: 事前フィルタリング

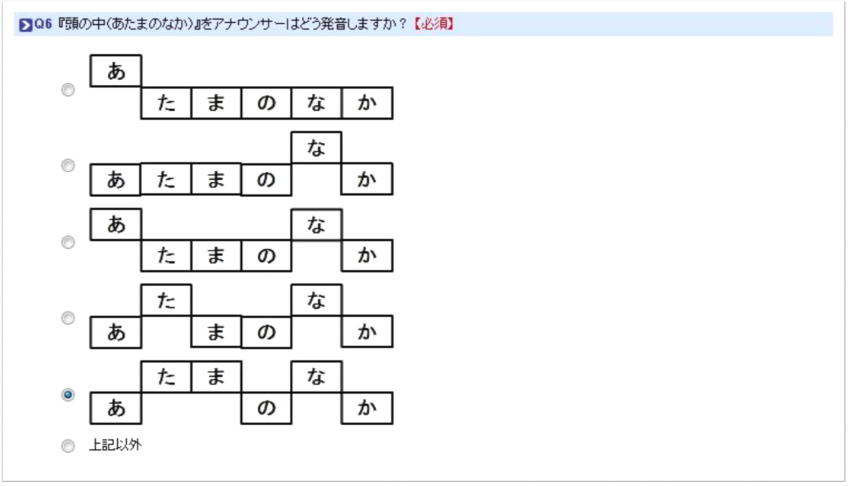


図 4.3: アクセント能力者を優先させるためのテスト例

表 4.1: 事前フィルタリングによるベースフィルタリング

Q1:あなたは現在，週に何時間くらい仕事をしていますか？	1.0 時間（仕事をしていない）
	2. 週に 1〜10 時間未満
	3. 週に 10〜20 時間未満
	4. 週に 20〜30 時間未満
	5. 週に 30〜40 時間未満
	6. 週に 40 時間以上
Q2:あなたは週に何時間くらい内職や副業をしても良いと思いますか？	1.0 時間（内職や副業はしない）
	2. 週に 1〜10 時間未満
	3. 週に 10〜20 時間未満
	4. 週に 20〜30 時間未満
	5. 週に 30〜40 時間未満
	6. 週に 40 時間以上
Q3:あなたが内職や副業をするにあたって，最低欲しい時給はいくらですか？	1. 400 円未満
	2. 400 円〜500 円未満
	3. 500 円〜600 円未満
	4. 600 円〜700 円未満
	5. 700 円〜800 円未満
	6. 800 円〜900 円未満
	7. 900 円〜1000 円未満
	8. 1000 円以上
Q4:簡単な内職に興味はありますか？	1. とても興味がある
	2. やや興味がある
	3. どちらでもない
	4. あまり興味がない
	5. まったく興味がない
Q5:あなたの最終学歴を教えて下さい	1. 中学卒
	2. 高校卒
	3. 専門大学卒
	4. 短大卒
	5. 大学卒
	6. 大学院（修士）卒
	7. 大学院（博士）卒
	8. 上記以外
Q6:学生時代に最も得意だった科目はなんですか？	1. 数学
	2. 国語

表は次ページに続く

前ページからの続き

	3. 理科
	4. 社会
	5. 音楽
	6. 家庭科
	7. 体育
	8. その他
Q7:スマートフォンを持っていますか？	1. iphone を持っている
	2. android スマートフォンを持っている
	3. スマートフォンは持っていないが 携帯を持っている
	4. 携帯もスマートフォンも持っていない
Q8:個人用のパソコンを持っていますか？	1. 個人用のパソコンを持っている
	2. 個人用はないが, 家族で共用のパソコンを持っている
	3. パソコンを持っていない

4.3 動的フィルタリング

ワーカーがタスク処理をしている際に行うフィルタリングである．事前フィルタリングにて最低限の品質を確保できたワーカーであるが，すべての低品質なワーカーを排除できたわけではない．また，人間は時間の経過に応じて能力が上下するため，初期の品質判定が継続するとは限らない．そのため，タスク処理を進めていく課程で動的にワーカーのフィルタリングを行うために正解率と経験値という 2 点の項目を設けている．

正解率は「正解数/総作業数」で算出し，一定値以下のワーカーはスパムワーカーとみなし，以降の PCSS におけるタスク処理を禁止する．正解率を求めるにあたって，それぞれの作業はあらかじめ正解が用意されていないため，ワーカーの処理結果が正解であるか否かを判定する手法が必要となる．この合否判定に用いる手段としては多数決を用いる手法が提案されている [Snow 08]．我々も主に多数決にて正解を決定しており，アンケートなど正解がない場合にはタスク説明に正解がない旨を明記し，正解率は変動させない．多数決が用いられるタスクは非常に多岐に渡るが，例として図 4.4 のような選択式のタスクが挙げられる．実際にこの作業をワーカーに処理させたところ，回答において 3 人が一致した率は 79.4%，2 人が一致した率は 19.0%，バラバラだった率は 1.3%，未回答は 0.3%となっ

た．そして3人が一致したケースに対して，作業結果から1000件をランダムで抽出，システム管理者側で手動で正解を作成して，正解率をチェックしたところ95.3%と非常に高い正解率を得ることが出来た．そのため，一般的なタスクの出題方法として，3人一致したケースのみデータとして採用し，残りのデータは不採用，もしくは再度クラウドソーシングに出題するというパターンを一般的に用いている．

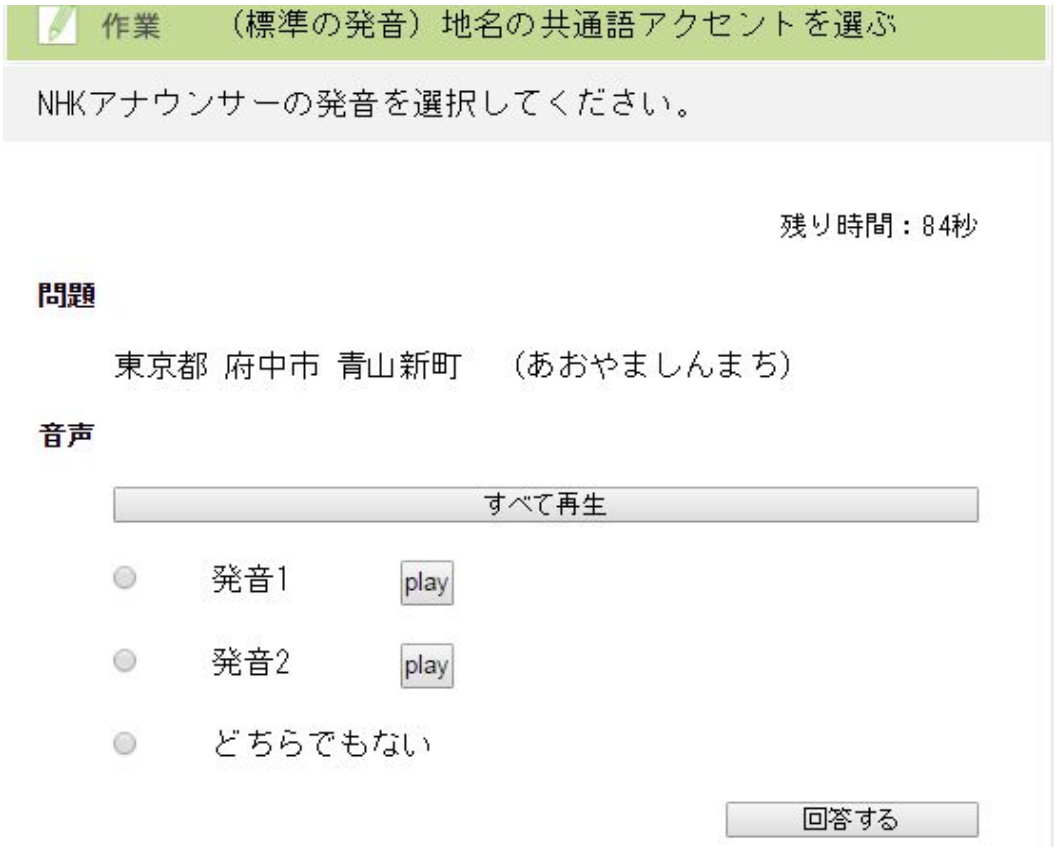


図 4.4: 多数決タスクの例

正解率が一定値以下になることでタスク処理ができなくなることはワーカーに明言してあり，ワーカーはこの数値表示によって精度に対する注意を喚起される．これらの数値は作業者の行動によって変化するという点でゲームメカニクスにおける得点制度と同等に考えることができる．得点制度はゲームメカニクスとしては一般的であり，利用者のモチベーションを向上させるための手法として用いられている [Ahn 08]．

また，同様に,「正解数 - 不正解数」で算出される経験値を設定し，一定の経験値を持つ

ワーカーに対して高報酬，高難易度のタスクを提供している．難易度の基準はリクエスタによって異なるが，多数決による正解判定を行なうタスクで結果が分散してしまう，タスクの完了まで時間がかかるなどのケースでは難易度が高いと判定される場合が多い．これらの数値は図 4.5 のように作業中に画面に常に表示している．

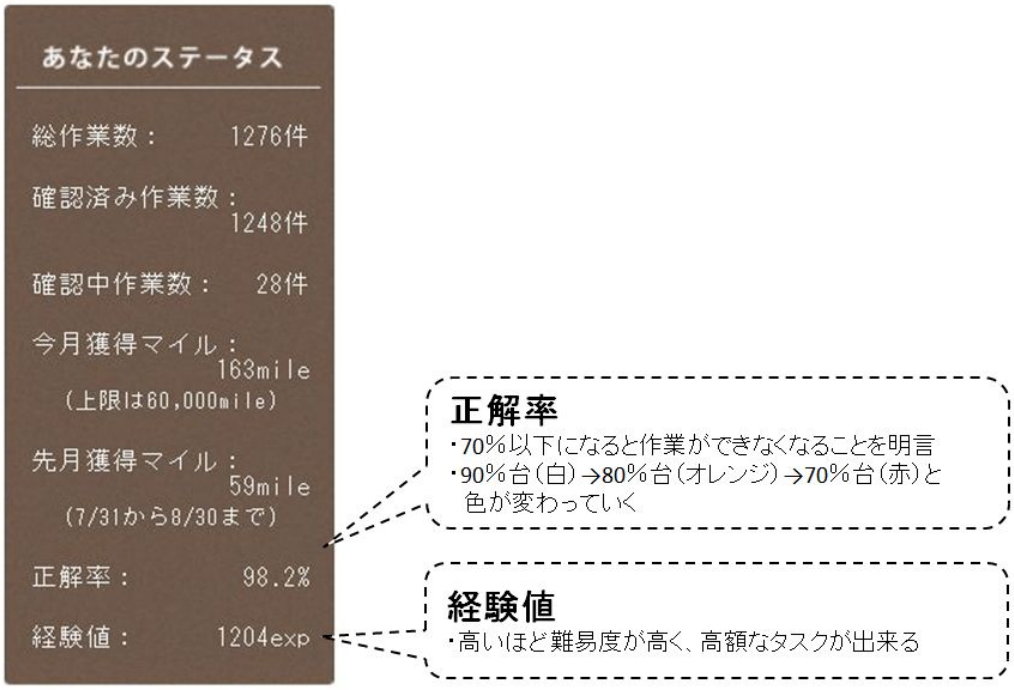


図 4.5: ワーカーに表示されるステータス画面

現時点ではインターフェイス上の制限から，ワーカーが確認することが出来るのはすべてのタスクの全体平均正解率のみである．しかし，動的フィルタリングをこの全体平均正解率のみで行うとフィルタリング効果が低いことがわかっている．我々は表 4.2 のようにカテゴリごとに正解率をワーカーに明示せず別途管理している．本研究では 8 章で事例として紹介した自然言語処理に関する「単語判定カテゴリ」「読み付けカテゴリ」「品詞カテゴリ」「アクセントカテゴリ」に関して述べる．表 4.2 を見ると正解率が低いカテゴリの作業総数が非常に少なく，正解率が高いカテゴリの作業総数が多いケースが散見される．これにより正解率が下がるような難易度の高いタスクをワーカーが避ける傾向があることがわかる．しかし，全体正解率のみで判断を行った場合,「賃金が高く難易度も高いタスク A」

と「賃金が低く難易度も低いタスクB」があった場合，ワーカーはタスクAを処理し，全体平均正解率が下がるとタスクBを行なって全体平均正解率を回復させるという行動をとることがあった．これは該当するカテゴリの正解率が低いにもかかわらず大量に作業を処理しており，かつ全体正解率は高いというワーカーの存在から判明した．これらのワーカーを低品質ワーカーと呼称し表4.3に数と割合（低品質ワーカー数/ワーカー数）を示す．このような低品質ワーカーの行動に対応するため特定カテゴリにおける作業総数が50以上になったワーカーをアクティブワーカーとし，特定のカテゴリにおけるアクティブワーカーの精度が一定値以下になった場合は，そのカテゴリに属するタスクを隠し，処理をさせないようにすることでワーカーの行動コントロールを行っている．PCSSでは該当カテゴリのタスクにおける精度が60%以下になったワーカーはその作業をさせないという対応をしている．これは70%以下のワーカーを排除対象とすると悪意の無いワーカーを多く排除してしまい，50%以下のワーカーを対象とすると，悪意のあるワーカーを排除しきれなかったというシステム運用上の経験からの数値である．

表 4.2: タスク別ワーカー結果精度（一部）

ワーカーID	読み付け			読み仮名判定			品詞判定		
	正解率	正解位数	不正解数	正解率	正解位数	不正解数	正解率	正解位数	不正解数
101	92.6	14368	1147	94.6	4295	244	88.6	474	61
102	96.2	53463	2086	97.1	23028	695	83.6	504	99
103	97.1	1455	43	98.3	5385	94	100.0	10	0
104	94.5	10247	597	95.8	28824	1250	83.9	3406	654
105	91.9	452	40	0.0	0	0	0.0	0	0
106	98.0	16010	329	99.3	11558	82	93.2	68	5
107	95.2	64775	3240	94.7	42631	2398	88.7	11915	1517
108	97.0	44965	1375	98.2	39815	741	89.4	1831	216
109	97.4	69290	1863	97.9	25541	543	84.4	1094	202
110	96.4	65581	2462	97.0	29294	903	90.5	7435	780
111	94.6	2164	124	95.7	44	2	100.0	13	0
112	90.4	64183	6792	95.0	52971	2814	82.4	17895	3833
113	95.9	77179	3313	96.1	94042	3841	89.8	13512	1532
114	95.5	121979	5746	95.9	85658	3629	47.1	3985	4483
115	90.0	98895	11040	97.0	114462	3533	92.7	11622	912

表 4.3: 各カテゴリにおける値

	作業数	ワーカー数	アクティブなワーカー数	低品質ワーカーの数(割合)
単語判定カテゴリ	1,652,271	454	353	50(11.0%)
読み付けカテゴリ	3,185,708	576	380	32(5.6%)
品詞カテゴリ	589,949	129	107	6(4.7%)
アクセントカテゴリ	1,270,618	358	276	33(9.2%)

4.4 結果フィルタリング

図 4.6 のように，ワーカーのタスク処理結果からワーカーの特徴を判別するフィルタリングである．動的フィルタリングは正解を判定することが出来る作業に対してのみ有効であり，アンケートや文章作成のような明確な正解がなく，多数決も実施しにくいタスクにおいては適用できない，また，ワーカーが低品質ワーカーであると判明し，出題停止に至るまでに多くの低品質なデータが算出されてしまうという欠点がある．我々はこの問題に対し，図 4.6 のように，ワーカーのタスク処理結果からワーカーの特徴を判別する結果フィルタリングを用いて対応している．

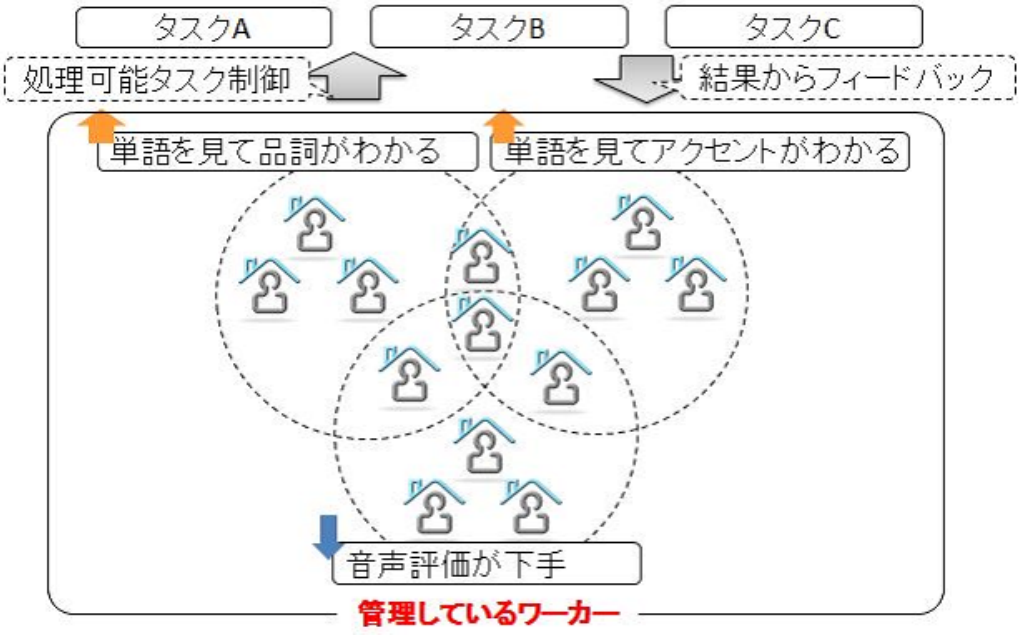


図 4.6: タスク処理結果を用いたワーカーの特徴付け

明確な正解がないタスクでも，リクエスタの意図に沿った内容か否かという判定は存在しており，この判定をリクエスタにタスク毎に行わせるには大きなコストがかかる．このようなタスクに関して，リクエスタは他のリクエスタの類似したタスクの結果や，小規模のテスト用タスクを実施した結果などから，正解率の高いワーカーや出題意図に沿った回答をしているワーカーを選別し，以降のタスクは条件に該当するワーカーのみに出題することで結果精度を向上させることができる．この選別基準はシステム側で明確に定めておらず，リクエスタによって異なる．これらのワーカーの情報で優秀なワーカーを判別するための情報を「スキル」，低品質なワーカーを判別するための情報を「負スキル」と呼称している.「負スキル」はカテゴリごとに作成可能であり,「負スキル」保持ワーカーは該当するカテゴリのタスク以外は作業できるため，全ての作業が不可能となっているスパムワーカーとは異なる．例えば「品詞」のカテゴリのタスクの正解率が高いワーカーには「品詞」のスキルを付与し,「品詞」のタスクは「品詞」スキルを持つワーカーにのみ出題することで精度向上を行っている．これらのスキルはリクエスタ間で共有して使用することが出来るため，新規のリクエスタも初回からスキルを保持するワーカーにタスクを処理させるこ

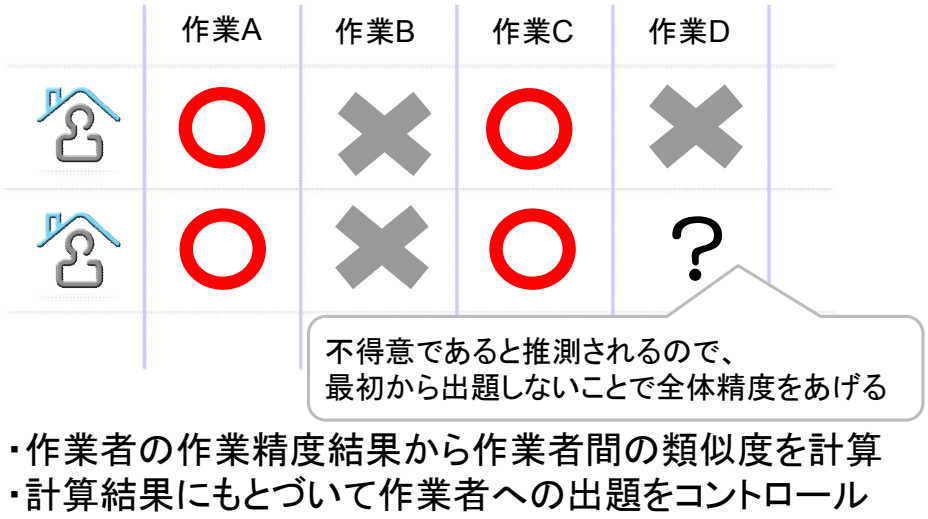


図 4.7: 推測フィルタリング

とが可能である.

4.5 推測フィルタリング

動的フィルタリングや結果フィルタリングは何らかのタスクの処理結果をワーカーの行動コントロールに流用したものであり，ワーカーがスパムワーカー，低品質ワーカーであった場合はワーカーの行動コントロールが出来る段階に達した時点で低品質な処理結果を残してしまっている事が多い．これらのデータは再処理が必要であり，大量のワーカーによって短時間で大量のタスク処理が行われるマイクロタスク型のクラウドソーシングでは時間，賃金ともに再処理のコストが大きくなってしまう．そこで，我々は更に低品質なタスク処理結果を削減するために，ワーカーの特性から行動を推測し，事前にタスクに不適切なワーカーをフィルタリングすることで精度向上を試みている（図 4.7）.

このようなワーカーに対するタスクの割り当てに関する研究として様々な研究がなされている．タスクの内容やワーカーのタスクに対する完遂率をベースにタスクの推薦を行なう研究 [Ambati 11] では低品質ワーカーに対する対応が取られておらずタスク推薦の効果があらわれるまでに多くの低品質データが発生してしまう問題がある．我々は推測フィル

タリングに至るまでの複数のフィルタリングで低品質ワーカーを可能な限り少なくすることで，低品質データの発生を最低限におさえている．また，ワーカーの行動履歴，ワーカーのタスクに対する嗜好からワーカーにタスクの推薦を行なう研究 [Yuen 12] でも対象となるワーカーが膨大になった場合のコストが大きいという問題がある．我々は前述のように推測フィルタリングに至るまでの複数のフィルタリングで対象となるワーカーの数を削減し，必要なコストを最低限に抑えている．また，タスクの難易度レベル，ワーカーのスキルのレベルを推測した結果からワーカーにタスクの推薦を行う研究 [Vaughan 13] でも対象となるタスクのカテゴリが限られているという問題がある．我々は複数のカテゴリを管理し，タスクをカテゴリに分類することで複数のタスクカテゴリを対象とすることを可能としている．

我々はワーカーの類似性を利用した協調フィルタリングを用いて，ワーカーが未作業のカテゴリのタスクの結果精度の推測を行い，精度が低いと推測されるカテゴリのタスクは最初から処理させないという方法を用いている．協調フィルタリングとは多くのユーザの嗜好情報を蓄積し，あるユーザと嗜好の類似した他のユーザの情報を用いて自動的に推論を行う方法である．協調フィルタリングにはコンテンツベースの協調フィルタリングとアイテムベースの協調フィルタリングが存在する．

コンテンツベースの協調フィルタリング

コンテンツベースの協調フィルタリングとは所有しているワーカーの情報（コンテンツ），すなわち年齢，性別，既婚，未婚，住所，学歴，労働意欲，得意な科目などの情報をベースに，ワーカーの類似性を判定して行うフィルタリングである．具体的な例として，表 4.4 のようなケースでは，次のようなパターンが考えられる．

(1) ワーカー a とワーカー c の類似度が高いのでワーカー a のタスク C に対する結果精度からワーカー c が未作業のタスク C の結果精度を推測する．ワーカー a のタスク C に対する結果精度が高いのでワーカー c のタスク C の結果精度も高く推測されるため，ワーカー c にはタスク C を積極的に勧める．

(2) ワーカー a とワーカー c の類似度が高いのでワーカー a のタスク A に対する結果精度からワーカー c が未作業のタスク A の結果精度を推測する．ワーカー a のタスク A に対す

る結果精度が低いのでワーカー c のタスク A の結果精度も低く推測されるため，ワーカー c にはタスク A を勧めない．

表 4.4: コンテンツベースの協調フィルタリングのデータ例（「-」部分は未作業）

	年齢	学歴	得意な科目	タスク A	タスク B	タスク C
ワーカー a	30 代	大卒	音楽	60%	-	90%
ワーカー b	60 代	短大卒	国語	-	-	90%
ワーカー c	30 代	大卒	音楽	-	-	-

アイテムベースの協調フィルタリング

アイテムベースの協調フィルタリングの具体的な例として表 4.5 のようなケースでは，次のようなパターンが考えられる．

(1) ワーカー a とワーカー b の類似度が高いのでワーカー a のタスク B に対する結果精度からワーカー b が未作業のタスク B の結果精度を推測する．ワーカー a のタスク B に対する結果精度が高いのでワーカー b のタスク B の結果精度も高く推測されるため，ワーカー b にはタスク B を積極的に勧める．

(2) また，ワーカー a とワーカー c の類似度も高いのでワーカー a のタスク E に対する結果精度からワーカー c が未作業のタスク E の結果精度を推測する．ワーカー a のタスク E に対する結果精度が低いのでワーカー c のタスク E の結果精度も低く推測されるため，ワーカー c にはタスク E を作業させない．

現在の PCSS ではワーカーの情報（コンテンツ）があまり多くないのでコンテンツベースの協調フィルタリングの信頼性が低い．一方でワーカーのタスク処理履歴が蓄積されるに応じてアイテムベースの協調フィルタリングは有効性を増すため，PCSS ではアイテムベースの協調フィルタリングを用いている．我々はユーザの嗜好情報の代わりにワーカーを特徴づける情報として，タスクのカテゴリ毎の結果精度を用いている．ワーカーをカテゴリ毎の結果精度のパターンで比較し，類似したワーカーの情報を用いて，未作業のカテゴリのタスクの結果精度の推測を行う．

表 4.5: アイテムベースの協調フィルタリングのデータ例（「-」部分は未作業）

	タスク A	タスク B	タスク C	タスク D	タスク E
ワーカー a	98%	95%	99%	-	50%
ワーカー b	99%	-	97%	-	60%
ワーカー c	98%	99%	90%	-	-

実際に我々が推測フィルタリングを行なうにあたって，必要なワーカーの類似度を計算するためにピアソン相関係数を用いている．ピアソン相関係数は協調フィルタリングにて類似度を判定する際に用いられることの多い値である．全ワーカーの集合を W，その要素を u, v，全タスクカテゴリの集合 T，その要素を i, j とする．この時あるワーカー u のタスクカテゴリ i における結果精度を $r_{u,i}$，ワーカー u の結果精度の平均を $\bar{r_u}$ とした場合，ワーカー u とワーカー v の類似度 $S_{u,v}$ は式 (4.1) のようになる．

$$S_{u,v} = \frac{\sum_{i \in T}(r_{u,i} - \bar{r_u})(r_{v,i} - \bar{r_v})}{\sqrt{\sum_{u \in W}(r_{u,i} - \bar{r_u})^2}\sqrt{\sum_{v \in W}(r_{v,i} - \bar{r_v})^2}} \tag{4.1}$$

式 (4.1) を用いて各ワーカーの類似度を計算した結果は表 4.6 のようになった．この結果よりワーカー間の類似度は一定ではなく，類似しているワーカーと類似していないワーカーが存在することがわかる．得られたワーカー間の類似度を元に，ワーカー u がまだ作業していないタスク i における予測タスク結果精度 $P_{u,i}$ は式 (4.2) のように計算することができる．

表 4.6: ワーカー間類似度（一部）

ワーカーID	101	102	103	104	105
101	1	0.43	−0.4	0.13	0.59
102	0.43	1	−0.07	0.76	0.58
103	−0.4	−0.07	1	−0.38	0.79
104	0.13	0.76	−0.38	1	0.51
105	0.59	0.58	0.79	0.51	1
106	0.31	0.92	0.24	0.62	0.51
107	−0.27	0.77	−0.54	0.86	−0.1
108	0.36	0.93	−0.26	0.68	0.11
109	0.73	0.86	−0.36	0.97	0.18
110	0.69	0.93	−0.23	0.82	0.59
111	−0.61	0.1	0.61	−0.39	0.24
112	0.18	0.56	0.21	0.38	0.23
113	0.1	0.46	0.16	0.04	−0.04
114	0.79	0.82	−0.44	0.97	0.49
115	0.11	0.07	0.67	0.29	0.78

ワーカーID

$$P_{u,i} = \bar{r_u} + \frac{\sum_{v \in W}(r_{v,i} - \bar{r_v})S_{u,v}}{\sum_{v \in W} \mid S_{u,v} \mid} \tag{4.2}$$

このようにして得られた予測タスク結果精度を元に，リクエスタによってタスクが出題されたタイミングでカテゴリの判定，カテゴリに応じた推測フィルタリングを実行する．その結果に基づきワーカーが得意と予想されるタスクをワーカーに優先的に提示し，不得意と予想されるタスクをワーカーに表示しないという方法で結果精度の向上を試みている．

推測フィルタリングにて協調フィルタリングを用いるに当たって，全員の正解率が高いタスクで発生した低品質ワーカーを推測できないという問題がある．この問題に対して我々は動的フィルタリング，結果フィルタリングで対応を行っている．

PCSS におけるこれら 4 つのフィルタリングは図 4.8 のように表すことができる．

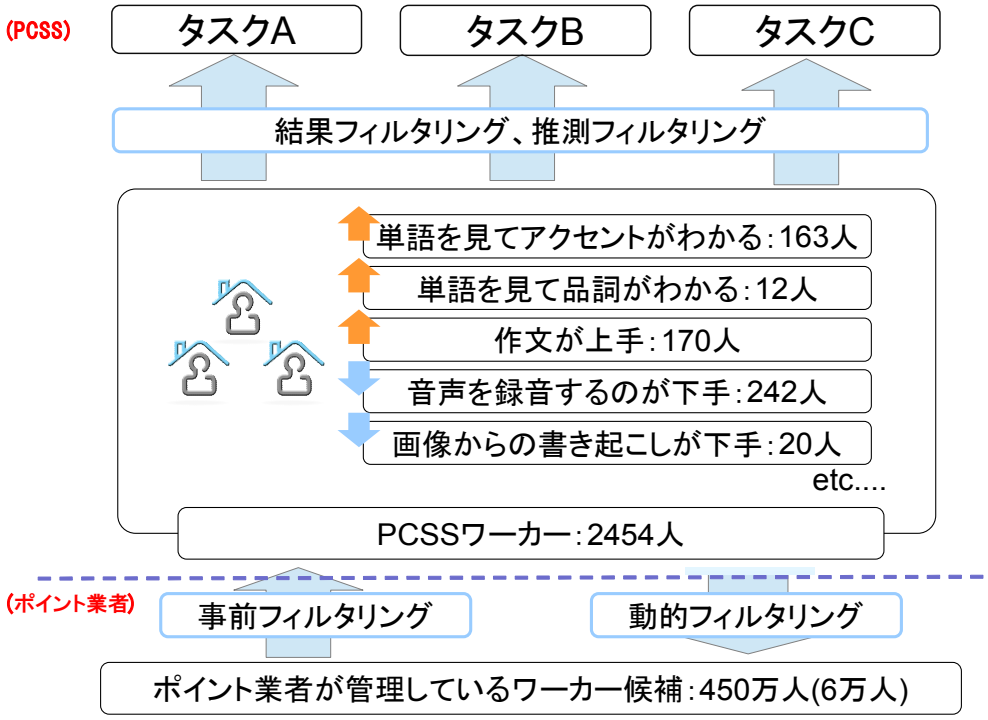


図 4.8: PCSS におけるフィルタリングの組み合わせ

第5章　ワーカーのフィルタリングによる精度向上手法の評価及び考察

本章ではPCSSを用いた語彙収集における4章で紹介したワーカーのフィルタリングによる精度向上効果に関して述べる．

5.1　事前フィルタリングの効果

ポイント業者の保持する会員のうち，8万人に対して事前フィルタリングとなるアンケートを行った．アンケートの項目は表4.1に示している．選別基準としては，一定以上の作業時間が確保できること，要求する時給は東京都の最低時給以上であること，モチベーションにつながる興味度合いが高いこと，学歴が短大卒以上であることなどを基準としている．マイクロタスク型のクラウドソーシングとしては，要求する時給が低いほどコストの削減につながるが，PCSSでは作業環境の向上ということも重視しているため，あまり時給を下げることは検討していない．このアンケートの結果からワーカー候補として2457人に絞込み，クラウドソーシングへの勧誘を行った．最終的に勧誘に応じてクラウドソーシングの作業を一度でも行ったワーカーは1630人（そのうち62人はスパムワーカーとして排除），実際に毎月実績のあるアクティブなワーカーは150人前後となっている．

5.2　動的フィルタリングの効果

PCSSでは動的フィルタリングにおいて排除すべき低品質ワーカーと判定する閾値を全体正解率70%以下としている．これは運用開始時に閾値を80%と仮設定した際に，不慣れなワーカーがすぐにスパムワーカー扱いされてしまいクレームが発生したためである．我々は結果精度を高く維持したい一方で，大量のタスクを高速に処理しなくてはならない．そ

のためにはワーカーをすぐに切り捨てるのは得策ではないと判断し，逐次閾値を下げていき，動的フィルタリング以外の精度向上手法を導入することで対応した．この全体正解率を用いた動的フィルタリングによって 1630 人のワーカーから 62 人のワーカーを低品質，スパムワーカーとして排除することができている．また，各カテゴリにおける低品質ワーカーの数と割合（低品質ワーカー数/ワーカー数）を表 5.1 に示す．これらの特定カテゴリにおける低品質ワーカーは該当するカテゴリに属するタスクが選択可能なタスク一覧から隠され作業ができなくなる．

表 5.1: 各カテゴリにおける値

	作業数	ワーカー数	アクティブなワーカー数	低品質ワーカーの数(割合)
単語判定カテゴリ	1,652,271	454	353	50(11.0%)
読み付けカテゴリ	3,185,708	576	380	32(5.6%)
品詞カテゴリ	589,949	129	107	6(4.7%)
アクセントカテゴリ	1,270,618	358	276	33(9.2%)

5.3 結果フィルタリングの効果

結果フィルタリングを用いることによって得られたスキルを表 5.2 に示す．単語判定，読み付けに関しては動的フィルタリングで排除されたワーカー以外のワーカーには大きな能力の差異は見られなかったため，スキル保持ワーカーの絞り込みは実施していない．動的フィルタリングと同様に，これらのスキルを元にタスク処理の可否を決定している．

表 5.2: 「スキル保持」「負スキル保持」と判定されたワーカ数

スキル名	対象タスクカテゴリ	ワーカー数
単語を見て品詞がわかる	品詞カテゴリ	31
発音の正誤判定ができる	アクセントカテゴリ	207
複数候補から正しい発音を選択できる	アクセントカテゴリ	142
単語を見て発音を記述できる	アクセントカテゴリ	53
音声を評価するにあたって問題がある	アクセントカテゴリ	242

品詞カテゴリタスクの作業結果を解析して作業結果が高品質であったワーカーには「単語を見て品詞がわかる」スキルが与えられる．リクエスタは難易度が高く精度を優先する品詞タスクを出題するときは「単語を見て品詞がわかる」スキル保持ワーカーのみに作業を出題して精度向上を行なう．

アクセントカテゴリに関しては複数の難易度の段階があり，リクエスタはその段階ごとにタスク化を行っている．さらにリクエスタは各難易度ごとにスキルを作成し，小規模タスクを行い，結果を人手でチェックして一定以上の正解率を持つワーカーに対してスキルを付与する．表 5.2 のアクセントカテゴリにおけるスキルは難易度が高い順で「単語を見て発音を記述できる」「複数候補から正しい発音を選択できる」「発音の正誤判定ができる」となっており，リクエスタは難易度が高いスキル（例：単語を見て発音を記述できる）を持つワーカーには難易度が低いスキル（例：複数候補から正しい発音を選択できる，発音の正誤判定ができる）を同時に付与する．つまり難易度が低いアクセント作業には高スキルワーカーから低スキル保持ワーカーまで全てに作業を行わせて処理速度を向上させ，難易度が高いアクセント作業には高スキル保持ワーカーのみに作業を行わせて処理速度を犠牲に精度を向上させる．また,「音声を評価するにあたって問題がある」スキルは負スキルであり，一番難易度の低いタスクにおいて結果品質が低いワーカーに付与されるスキルである．このスキルが付与されたワーカーにはアクセントカテゴリに属するタスクの処理をさせないことで精度向上を行なう．

5.4 推測フィルタリングの効果

実際に推測フィルタリングを行うにあたって，式 (4.2) で得られた予測タスク結果精度 $P_{u,i}$ の精度を確かめるために，今までの PCSS の運用データを用いて実験を行った．各ワーカーの結果精度をカテゴリ毎に集計し（図 4.2），その集計結果を元にピアソン相関係数を用いてワーカーの類似度を計算した．図 4.2 で既に実際の解答履歴から算出されているタスク i におけるワーカー u の実測タスク結果精度 $M_{u,i}$ と，他のワーカーとの類似度から推測した予測タスク結果精度 $P_{u,i}$ を比較検証した.「品詞カテゴリ」を例に用いた場合，得られた実測タスク結果精度 $M_{u,i}$ と予測タスク結果精度 $P_{u,i}$ の比較は図 5.1 のような結果となる．各カテゴリにおける実測タスク結果精度と予測タスク結果精度の値の差の平均，予測

タスク結果精度が90%以上のワーカーを推測高精度ワーカーと呼称し，その人数，推測高精度ワーカーの実測タスク結果精度を調査し，実際に結果精度が90%以上であるワーカーの数を推測高精度ワーカー正解数と呼称し，その人数を表5.3に示す．

表 5.3: 実測タスク精度と予測タスク精度の比較

	推測値誤差	推測高精度ワーカー数	推測高精度ワーカー正解数
単語判定カテゴリ	4.44	183	163
読み付けカテゴリ	3.69	219	194
品詞カテゴリ	4.45	23	23
アクセントカテゴリ	4.27	138	121

効果を確認するために，各カテゴリに対して精度向上適用前と適用後それぞれのタスクの処理結果から無作為に各カテゴリごとに1000件のデータを抽出し，人手によって合否を確認することで精度を計測した．対象となるデータは実務上の測定であるため同一の問題ではないが，同一条件で行ったWebクローリングによって取得した125億文のWebテキストデータに対して，同一の辞書で形態素解析を行い，得られた未知語候補22万語を単語判定，読み付け，品詞付け，アクセント付けの各カテゴリにおけるタスクで処理した結果のデータである．この収集に関する詳細は8章で述べる．結果を表5.4に示す．このように複数の精度向上手法により，実際に研究データに利用可能なデータの取得効率が向上していることがわかる．

表 5.4: 各カテゴリにおける精度向上効果

	精度向上適用前正解率	精度向上適用後正解率
単語判定カテゴリ	65.9%	89.6%
読み付けカテゴリ	56.3%	94.0%
品詞カテゴリ	71.0%	90.4%
アクセントカテゴリ	54.1%	98.7%

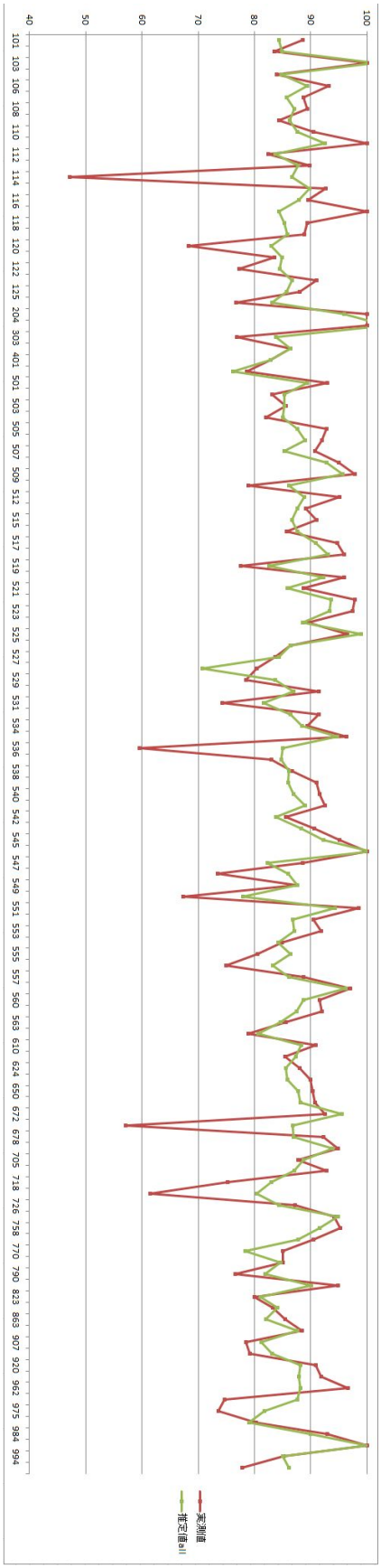


図 5.1: 実測タスク結果精度 $M_{u,i}$ と予測タスク結果精度 $P_{u,i}$ の比較

実際に読み付与タスクにおいてはPCSSの精度向上手法を用いずに処理した場合，研究データに利用可能な精度を持つ処理結果は手動でチェックした1000件のうち563件(56.3%)と低かったが，精度向上手法を用いることによって最終的に手動でチェックした1000件のうち940件(94.0%)が研究データに利用可能な精度を得た．また，アクセント付与タスクに関しても精度向上手法適用前は研究データに利用可能な精度を持つ処理結果は全体の54.1%と低かったが，精度向上手法適用後は全体の98.7%が研究データに利用可能な精度を得た．また，各カテゴリにおける精度向上適用前，精度向上適用後の各5000件のデータに対して「作業を行った高精度アクティブワーカー数，高精度非アクティブワーカー数」「作業を行った高精度ワーカー以外のアクティブワーカー数，高精度ワーカー以外の非アクティブワーカー数」を表5.5に示した．

表 5.5: 各カテゴリにおけるワーカー数

	精度向上手法適用前			
	高精度ワーカー数		高精度ワーカー以外のワーカー数	
	アクティブ	非アクティブ	アクティブ	非アクティブ
単語判定カテゴリ	15	0	8	0
読み付けカテゴリ	22	0	1	1
品詞カテゴリ	12	0	16	0
アクセントカテゴリ	6	0	1	0
	精度向上手法適用後			
	高精度ワーカー数		高精度ワーカー以外のワーカー数	
	アクティブ	非アクティブ	アクティブ	非アクティブ
単語判定カテゴリ	33	1	17	9
読み付けカテゴリ	51	4	8	5
品詞カテゴリ	12	0	0	0
アクセントカテゴリ	8	0	0	0

用いたデータは表5.4で用いたデータと同一条件で抽出した．単語判定カテゴリや読みつけカテゴリに関しては難易度が低く，リクエスタから処理速度が優先とされているため結果フィルタリング，推測フィルタリングは用いていない．そのため精度向上手法適用前，精度向上手法適用後ともに高精度ワーカーと通常ワーカーが混在して作業を行っている．その後，高精度ワーカー以外のワーカーのうち低品質ワーカー（結果精度70％以下）は動的

フィルタリングで排除されるため，表5.4で示したように事前フィルタリング，動的フィルタリングで十分な精度向上効果を得ることができている．品詞カテゴリやアクセントカテゴリに関しては難易度が低く，リクエスタから精度が優先とされているため結果フィルタリング，推測フィルタリングを用いて精度改善を試みた．高精度ワーカー以外のワーカーは結果フィルタリング，推測フィルタリングで事前に排除されるため，精度向上手法適用後は作業をすることはない．また，高精度ワーカーのみが処理を行なうため，ワーカー数が制限され，処理速度が低下するが，リクエスタから精度が優先とされているため，速度に関しては問題視されていない．

5.5 考察

以上のように，我々は研究データの作成にPCSSを用いるにあたって，結果データの品質を重視している．初期のPCSSでは事前フィルタリングと動的フィルタリングを用いて運用していたが，得られたタスク処理結果は研究データとして利用できるデータとして満足行くものではなかった．PCSSを運用していく過程でタスクのカテゴリ管理，結果データの解析などを導入することで，ワーカーには画一的なスパムワーカーだけではなく，特定のカテゴリが得意なワーカー，不得意なワーカーが存在することがわかってきた．これらのワーカーは自らの得意不得意を意識せず，報酬や興味に応じて作業を行なうため，結果として低品質な結果データの算出につながっている．しかし，これらのワーカーはスパムワーカーと異なり適当な回答や適当な入力を行なうという悪意のある行動は少なく，適切なコントロールを行なうことで得意分野を活かすことができると判断した．その結果，結果フィルタリング，推測フィルタリングの導入に至り，現在は精度の高い結果データを得ることが可能となっている．

このように本章では低品質なワーカーを排除するフィルタリングを中心に行った．一方で低品質なワーカーが作業を継続することによってスキルが向上し，高品質ワーカーとなるケースも存在し，そのようなケースを有効に活用して精度を向上させる手法が必要となる．そのようなワーカーの育成に関して次章以降で述べる

第6章　ワーカーの段階的学習による精度向上手法の提案

本章ではマイクロタスク型クラウドソーシングにおける段階的学習手法に関して述べる．まずはマイクロタスク型クラウドソーシングにおけるワーカーの育成及び学習の必要性に関して述べ，そのための手法であるタスクグループのタスクカテゴリ分類と，タスクカテゴリ間の関係性の解析に関して述べる．

6.1　クラウドソーシングにおける学習の必要性

クラウドソーシングでは不特定多数のワーカーが大量の作業を処理しているため，ワーカーも様々な人材が参加している．特に 4.1 節で述べているスパムワーカーと呼ばれるワーカーは適当な入力やスクリプトによる自動化等で結果精度を大きく低下させる原因となるため，早急な排除が必要となる．スパムワーカーは悪意の作業者であるため，PCSS では排除することに社会通念上，問題はないと考えている．しかし，善意の作業者かつ，品質が良くない作業者のケースでは安易な排除は望ましくない．これはクラウドソーシング市場の今後の成長に伴い，クラウドソーシングという労働形態が一般的になった場合に安易なワーカーの排除はクラウドソーシングの市場へのワーカーの参加を減少させ，市場の成長を妨げかねないからである．このような善意の作業者かつ品質が良くない作業者は，多くのタスクではタスクの処理精度は良いが，特定のタスクではタスクの処理精度が低いワーカーと捉えることができる．各タスクカテゴリ間での精度の相関性を図 6.1 に示す．この図ではそれぞれのタスクカテゴリにおける精度を x 軸，y 軸に設定し，特定のタスクでは精度が良いが，特定のタスクでは精度が悪いワーカーの存在を確認している．図において点線の円で示しているように，そのようなワーカーが実際に存在することがわかる．そのようなワーカーの例を表 6.1 に示す．

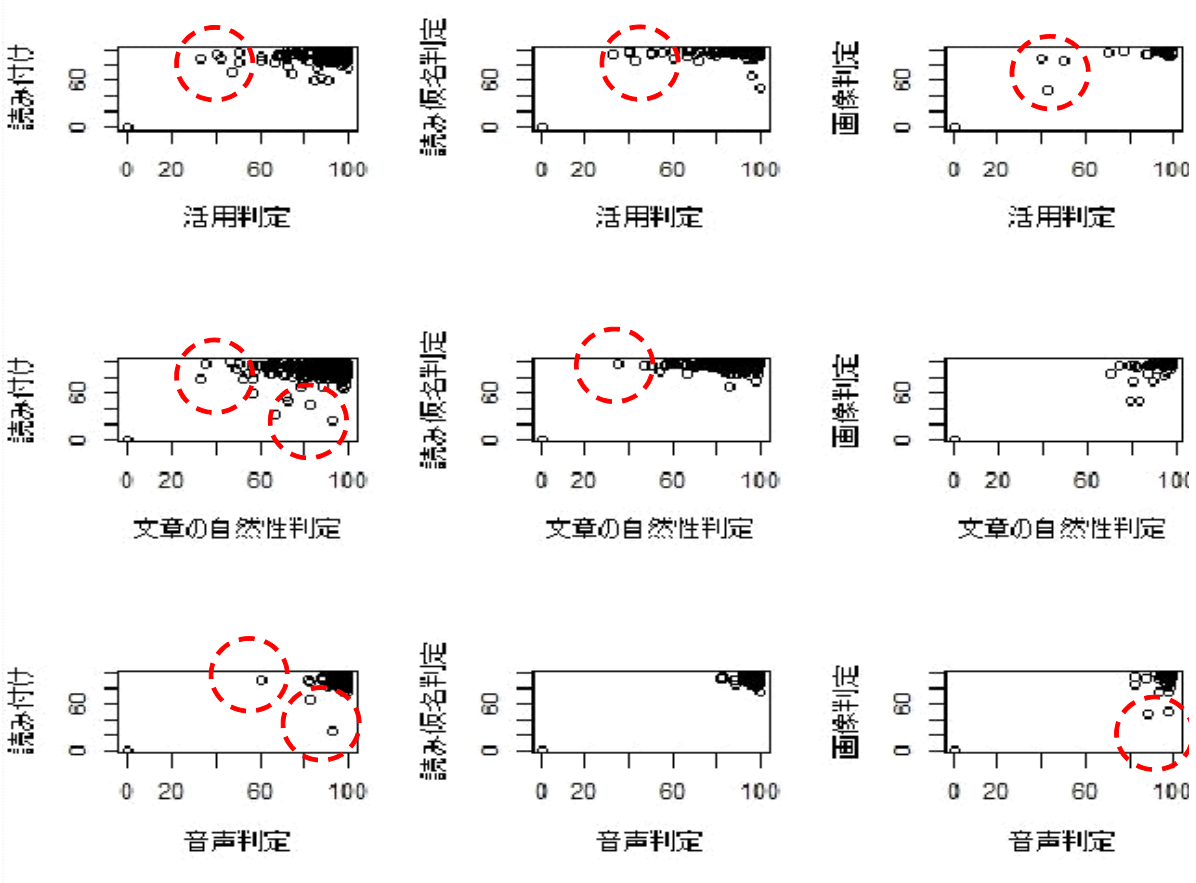


図 6.1: タスクカテゴリごとのワーカーの精度の相関性（一部）

表 6.1: 特定のタスクで精度が悪いワーカーの例

ワーカID	読み付け			読み仮名判定			画像判定			品詞判定		
	正解率	正解位数	不正解数	正解率	正解位数	不正解数	正解率	正解位数	不正解数	正解率	正解位数	不正解数
109	97.4	69290	1863	97.9	25541	543	97.4	38	1	84.4	1094	202
110	96.4	65581	2462	97.0	29294	903	0.0	0	0	90.5	7435	780
111	94.6	2164	124	95.7	44	2	0.0	0	0	100.0	13	0
112	90.4	64183	6792	95.0	52971	2814	96.0	3308	138	82.4	17895	3833
113	95.9	77179	3313	96.1	94042	3841	94.0	219	14	89.8	13512	1532
114	95.5	121979	5746	95.9	85658	3629	100.0	3	0	47.1	3985	4483

他の作業は精度が高いのに
この作業は苦手

これらの特定のタスクで精度が悪いワーカーは 4 章で説明したフィルタリングで該当す

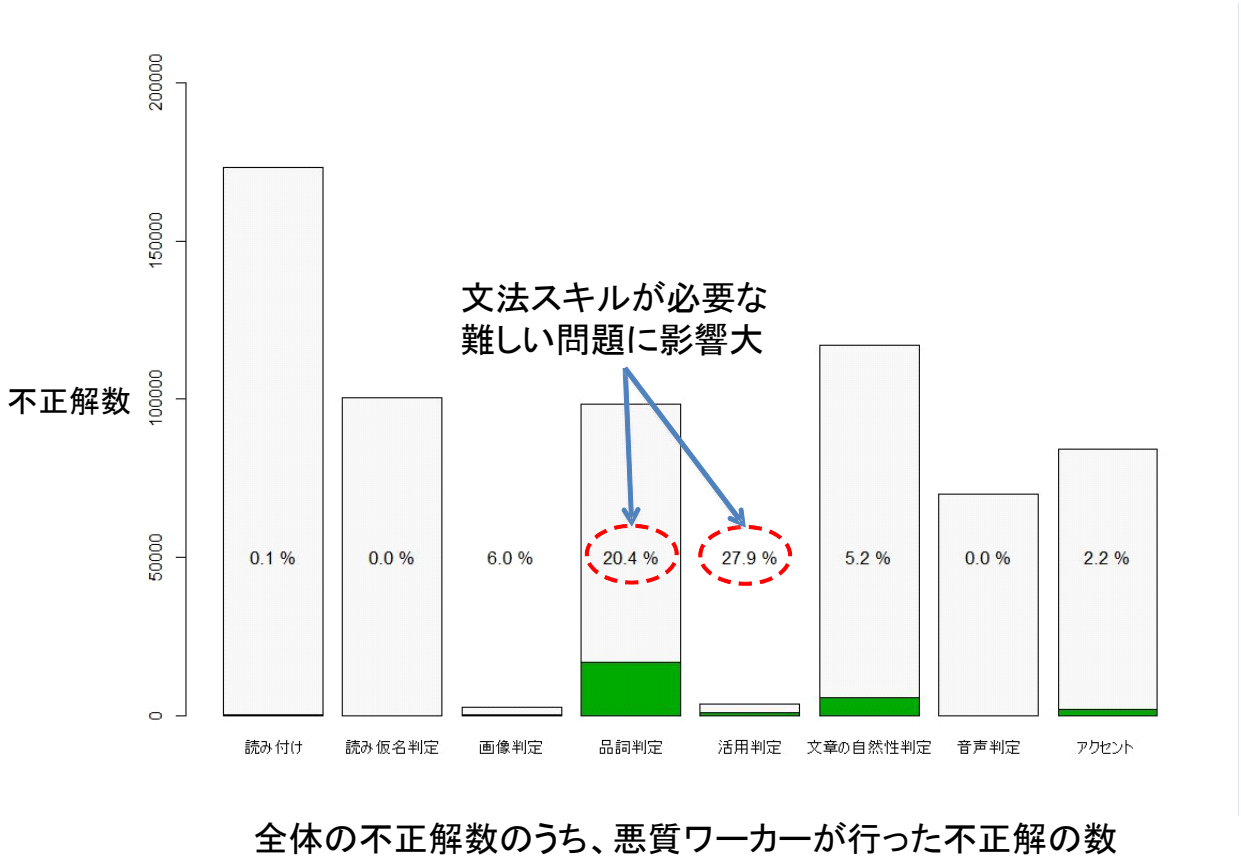


図 6.2: 低品質ワーカーが与える悪影響

るタスクから排除している．しかし，難易度が高い等で処理できるワーカーが少なく，処理速度向上のために意図的に排除しないケースなどでは図 6.2 に示すように，特定のタスクで精度が悪いワーカーが作成する不正解データの割合は大きなものとなり，結果としてリクエスタが無駄なコストを支払う結果となっていた．

このような特定のタスクで精度が悪いワーカーとスパムワーカーとの違いは悪意の有無である．悪意のあるワーカーはこちらのコントロールを無視してしまうのでコントロールの意味がないが，悪意のないワーカーならば，適切な学習を行わせるなどコントロールすることで精度を向上させることが可能なのではないかと考えた．PCSS では 3.3 節で述べたトレーニングフェーズでタスクの解説を行っているがこの作成はリクエスタ依存であるた

め，システム側でコントロール可能な学習方法として段階的学習方法を提案する．

6.2　段階的学習法の提案

学習とは一般的に一定場面でのある経験が，その後，同一または類似の場面での行動に良い変容をもたらすことということができる．これを様々な分野の作業に当てはめた場合，特定の作業で良い結果を得るためには同一，または類似の作業での経験を積まなければならないということができる．しかし，ある作業において作業者が作業内容を学習するにあたっては，最初から対象となる作業と同等の難易度の作業で学習を行うのではなく，目的の作業に関連する難易度の低い作業から開始して訓練し，段階的に難易度を上げていくことで作業者の能力を向上させていく段階的な学習方法をとることが一般的である．この手法は学校教育の仕組みと同じであり有効であることは示されている．しかし，学校教育は先生という熟達した管理者によって，様々な学習内容の目的や内容に応じた様々な科目への振り分け，今までの教育経験に裏付けされた学習カリキュラムの設計などがなされ，膨大な学生が実際に定められた学習方法を行い，その結果をフィードバックすることで高い効果を保証しているという点がある．このように段階的学習方法は非常に効果的であるが，段階的学習方法をクラウドソーシングのタスク処理に適用した場合，先生役の不在が問題となる．マイクロタスク型のクラウドソーシングは対象となるタスクが多岐にわたっており，ワーカーも不特定多数であるため，これらに対して学校教育のように明確な科目分け，学習カリキュラムの構築を手動で行う先生役を負担することはリクエスタにとってもシステム管理者にとってもコストの面で現実的ではない．そのため現状多くの場合は目的のタスクを説明するための単純な練習画面を手動で作成するにとどまっている．

我々はこの問題を解決するために従来のワーカーの行動履歴を解析して，学習内容の科目への振り分けと同様に，タスクの目的や内容に応じた様々なカテゴリへの振り分け，学習カリキュラムの設計と同様に，ワーカーへの最適な学習タスクカテゴリの割り当てを行う手法を提案する．また，学習タスクカテゴリを手動で作成するのはコストが高いため既存のタスクカテゴリを再利用して学習タスクカテゴリとして使用する方法を提案する．

前述の学習の定義に従えば，タスクの処理結果の精度を向上させるためには同一，または類似のタスクでの経験を積まなければならない．つまり，タスク A を実施してからタス

クBを実施した場合と，タスクAを実施せずにタスクBを実施した場合で，多くのワーカーが前者のケースでタスクBの処理結果が向上していた場合，タスクAはタスクBの練習用タスクとして扱うことができる．この考えに従って，今までのワーカーの行動履歴を解析し，タスクBに対する最適な練習用タスクAを見つけ出すことが我々の提案である．この方法は大量のワーカーの行動履歴とタスクが必要となるため難易度が高かったが，PCSSでは多くの運用実績を持っており，これらの大規模データを利用することで実現が可能となった．

人間の教師を必要とせずに学習者へと最適なカスタマイズを行い，かつ学習者から効果的なフィードバックを得ることを目的としてITSが提案されている [Ueno 00]．ITSは段階的学習手法を用いるにあたって非常に効果的な手法であるとみなされており，プログラミング学習など様々な分野で活用されている [Butz 06]．従来，ITSでは述語論理表現に基づく知識表現が一般的であったが，述語論理表現では，ルールの例外に対する処置が難しく，学習者の矛盾した反応に対する柔軟な対応が難しいなどの問題があった．例えば，タスク「単語かどうかの判定」に正解できることがタスク「漢字の読みの入力」に正解できるという必要条件であるというルールが存在した場合，実際にはタスク「単語かどうかの判定」を間違えるが，タスク「漢字の読みの入力」に正解するというケースがケアレスミスや当て推量によってありうる．これはワーカーが不特定多数であり品質が一定していないマイクロタスク型のクラウドソーシングにおいては顕著である．しかし，述語表現によるルール表現ではこのようなケースを処理することは非常に難しい．そこで，ベイジアンネットワークによる確率的アプローチによって，このようなケアレスミスや当て推量を確率に組み込み，矛盾したデータについて合理的な推論を行う．ベイジアンネットワークは因果的な特徴を有向グラフとして表す，現象の因果性，連関性を計算的に推論する理論・技術であり，PCSSでは6.3節で得られるタスクカテゴリ間の関係性を解析するために用いる．

ITSでは様々な学習要素をベイジアンネットワークのノードとして定義し，学習要素の関連性を解析している．ここで述べる学習要素とは「四則演算」や「一次方程式」などの項目であり，手動で定義されている．しかし，PCSSで扱うタスクは多種多様であり，同様にベイジアンネットワークのノードとしてタスクを用いた場合は計算量の面で現実的ではない．そのため，我々は大量のタスクグループを内容ごとにカテゴリに分類し，その後，

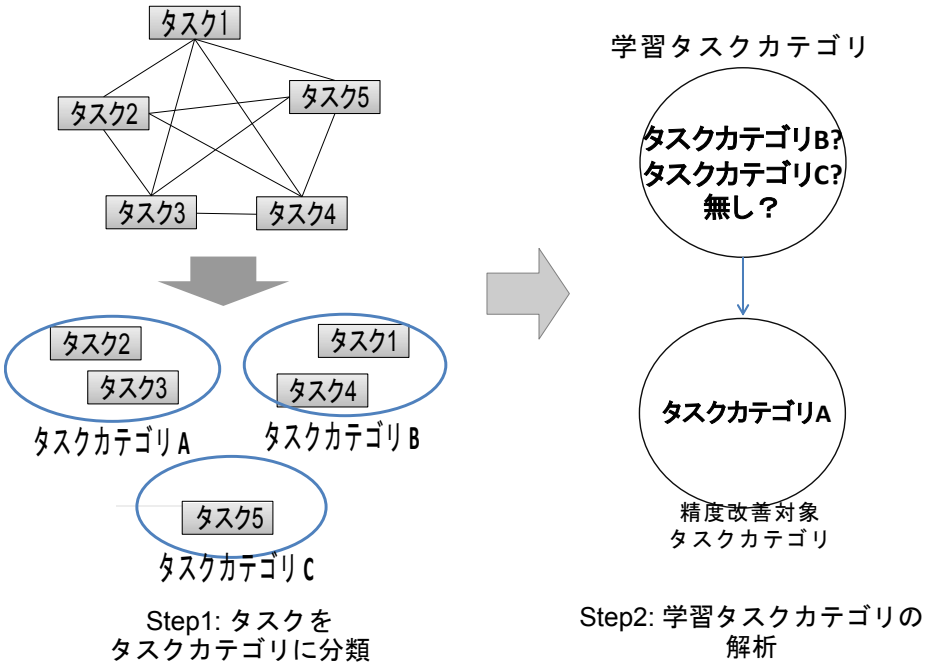


図 6.3: 学習タスクカテゴリ導出のためのステップ

得られたタスクカテゴリ間の関連性の解析を行うことで，学習タスクカテゴリを導出する．このステップを図 6.3 に示す．

6.3　STEP1: タスクグループのカテゴリ分類

PCSS では同内容のタスクを出題する際にはリクエスタが同内容のタスクをまとめてタスクグループとし，タスクグループのタイトル，説明文を付与して出題する．タイトル，説明文はシステム管理者がチェックを行い，不適切と思われる内容はリクエスタに再考を依頼している．異なる内容のタスクは別タスクグループとして出題される．本研究ではタスクグループをカテゴリ化するために，タスクグループのタイトルと説明文を形態素解析し，得られた単語を元に各タスクグループの TFIDF 値を計算する．その後に得られた TFIDF 値を用いて各タスクグループ間の類似度を計算して類似度の高いタスクグループ同士をカテゴリ分類する．タスクグループ t における単語 i の出現回数を $W_{t,i}$，タスクグループ t におけるすべての単語の出現回数の和を $W_{t,all}$，全てのタスクグループ数を T_{all}， 単語 i の出

現するタスクグループ数を T_i とした場合，タスクグループ t における単語 i の TFIDF 値 $TFIDF_{t,i}$ は式 (6.1) のように計算することができる．

$$TFIDF_{t,i} = \frac{W_{t,i}}{W_{t,all}} \log \frac{T_{all}}{T_i} \tag{6.1}$$

得られた各タスクグループにおける各語彙の TFIDF 値を用いて，各タスクグループ間における類似度の計算を行った．類似度の計算にはコサイン類似度を用いており，タスクグループ t における単語 i の TFIDF 値を $TFIDF_{t,i}$，全単語の集合を W とした場合，タスクグループ $t1$ とタスクグループ $t2$ 間のコサイン類似度 $cos(t1, t2)$ は式 (6.2) のように計算することができる．

$$cos(t1, t2) = \sum_{i \in W} TFIDF_{t1,i} \cdot TFIDF_{t2,i} \tag{6.2}$$

その後，得られた類似度を用いてタスクのカテゴリ分類を行う．カテゴリ分類のアルゴリズムは 1) 分類対象となるタスクグループを各カテゴリ所属のタスクグループ全てと比較し，最も類似しているタスクグループが所属するカテゴリに分類，2) 閾値を定め，どのカテゴリのどのタスクグループとも類似度が閾値以下なら新カテゴリを割り当てる，の繰り返しである．

現状の PCSS における 1853 万個のタスク，4153 個のタスクグループに本手法を適用した．全タスクグループ間の類似度を得るために，4153 タスクグループ間の対の全組み合わせ 1724 万通りに対して類似度の計算を行い，得られた類似度を用いてタスクのカテゴリ分類を行うことで 138 個のタスクカテゴリに分類することができた．この適用ではコサイン類似度 0.4 を閾値としている．閾値を求めるにあたって，複数の閾値候補でタスクのカテゴリ分類を行い，それぞれの結果における 100 件を手動で確認した (表 6.2)．その結果，コサイン類似度 0.4 以上の場合は別カテゴリに所属すると思われるタスクグループは存在しなかったことから，コサイン類似度 0.4 を閾値としてタスクのカテゴリ分類を行った．

表 6.2: タスク間類似度

コサイン類似度	別カテゴリと判定された数
0.0 以上 0.1 未満	88
0.1 以上 0.2 未満	43
0.2 以上 0.3 未満	17
0.3 以上 0.4 未満	4
0.4 以上 0.5 未満	0
0.5 以上 0.6 未満	0
0.6 以上 0.7 未満	0
0.7 以上 0.8 未満	0
0.8 以上 0.9 未満	0
0.9 以上 1.0 未満	0

6.4　STEP2: タスクカテゴリ間の関係性の解析

クラウドソーシングにおけるタスク処理結果にベイジアンネットワークを用いるにあたって，タスク A における処理結果 $T(A)$ が高精度である確率を $P(T(A))$，タスク B における処理結果 $T(B)$ が高精度であった場合にタスク A における処理結果 $T(A)$ が高精度である条件付確率を $P(T(A) \mid T(B))$ のように表すと，$P(T(A) \mid T(B))$ が高確率ということは「タスク B の処理精度が高かったときにタスク A の処理精度が高い確率」が高確率で発生するということであるため，タスク B を学習タスクとして扱うことができると仮定している．$P(T(A) \mid T(B))$ は式 (6.3) のように計算することができる．

$$P(T(A) \mid T(B)) = \frac{P(T(B) \mid T(A))P(T(A))}{P(T(B))} \tag{6.3}$$

このようにクラウドソーシングワーカーの行動履歴をベイズの定理を用いて解析することにより，精度向上させたいタスクのための学習タスクを生成する．ベイジアンネットワークの操作は大別して 1) ベイジアンネットワークの学習，2) ベイジアンネットワークを用いた推論の 2 つで行われる．

ベイジアンネットワークをクラウドソーシングに適用した具体的な例をあげる．クラウドソーシングにおけるタスクとワーカーの行動履歴からベイジアンネットワークの学習を

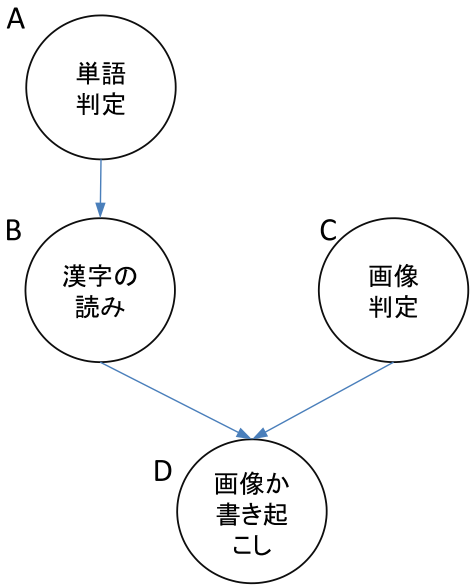


図 6.4: ベイジアンネットワークをクラウドソーシングに用いた例

行い，図 6.4 のような有向グラフが得られたと仮定する．タスク A はタスク B に影響し，タスク B，C はタスク D に影響することがわかる．すなわち，タスク B，C を処理した後にタスク D を処理したタスク処理結果精度と，タスク B，C を処理せずにタスク D を処理したタスク処理結果精度を比較した場合，前者の方がタスク処理結果の精度が高かった場合，タスク B，C をタスク D の学習タスクとして取り扱うことで結果精度の向上を狙う方法が考えられる．

ベイジアンネットワークの学習は与えられた学習データからベイジアンネットワークの解候補を作成し，ベイジアンネットワークの評価を行い，必要に応じて新たなベイジアンネットワークの解候補を構築するという作業で行われる [本村 11]．本研究ではベイジアンネットワークを学習，構築するに当たって Weka (Waikato Environment for Knowledge Analysis) 3.6.11 [1] を用いた．Weka で条件付確率表を作成するにあたって，6.3 節で得られた 138 個の各タスクカテゴリにおけるワーカーの平均精度を用いた（表 6.3）．各ワーカー毎にワーカーの行動履歴から各タスクカテゴリの平均正解率が 90%以上である確率を計算し，Weka の arff フォーマットのファイルを作成して Weka で使用している．ワーカーによっ

[1] http://www.cs.waikato.ac.nz/ ml/weka/

ては特定のタスクを実施していないワーカーもおり，この場合は欠損値としてあつかった．Weka では各ノードに対して，親候補となるノードの集合を作成，子ノードごとに親ノードと条件付き確率を決定し，その結果から最適な局所木を構築することで，最終的に最適なベイジアンネットワークを構築している．この処理過程において発生する局所木は膨大な数になるため，現実的なコストで最適なベイジアンネットワークを見つけ出すためには様々な手法が存在する．Weka では HillClimber，TabuSearch，Simulated Annealing，genetic Search など様々な探索アルゴリズムを選択することが可能である．本研究では「TID 0:読点の位置が正しいか判定」をサンプルとして用い，それぞれの探索アルゴリズムを用いて実験したところ simulated annealing(焼きなまし法) が最も適した有向グラフを得ることができたため，以降の実験における探索アルゴリズムとして simulated annealing を用いている．また Weka，および simulated annealing におけるパラメタを設定するにあたって，学習タスクが生成できる有向グラフが得られること，学習タスクが多くなりすぎると学習のためのコストが大きくなるため学習タスクが多くなりすぎないことの 2 点を前提として調整を行った．その結果マルコフブランケット分類器は用いず，10 分割交差検証を実施，simulated annealing のパラメタは温度（時間と共に変化するグローバルなパラメタ）の初期値（Tstart）は 10.0，温度の変化度合（delta）は 0.999，結果がマルコフブランケットになるようにはせず（Markov Blanket Classifier は false）, 反復回数（runs）は 10000，評価指標（scopeType）は BAYES，乱数の初期値（Seed）は 1 とした．また，このパラメタの調整にあたっても「TID 0:読点の位置が正しいか判定」をサンプルとして用いている．パラメタの調整においては Markov Branket Classifier の変更では有向グラフに変化は無く，TStart は小さすぎると学習タスクの作成に失敗する傾向があり，大きすぎると大量に学習タスクが発生する傾向があった．また，runs や seed は小さくしすぎると失敗する傾向があった．最終的に前述のパラメタで表 6.5 に示す学習タスクが得られ，表 7.1 で示す学習効果が得ることができたため，その他のタスクカテゴリに関しても同様のパラメタを用いた．また，本適用において各タスクカテゴリにおける作業タスク数が 50 以下のワーカーは作業量が少ないため平均精度としては用いず，そのタスクカテゴリにおけるタスクは処理していないものとして扱った．

本手法を適用して有向グラフを作成するにあたり，各タスクカテゴリにおけるワーカー

の平均精度を用いた（表 6.3）．ワーカーの行動履歴から各タスクカテゴリの平均正解率が 90%以上である確率を計算し，任意のタスクカテゴリ X の平均正解率 $T(X)$ が 90%以上であった場合にタスクカテゴリ A におけるタスクの平均正解率 $T(A)$ が 90%以上である確率 $P(T(A) \mid T(X))$ を用いて得られた有向グラフではタスクカテゴリ X はタスクカテゴリ A の学習タスクカテゴリとして扱うことができることが示される．

表 6.3: タスクカテゴリごとのタスク処理結果精度（一部）

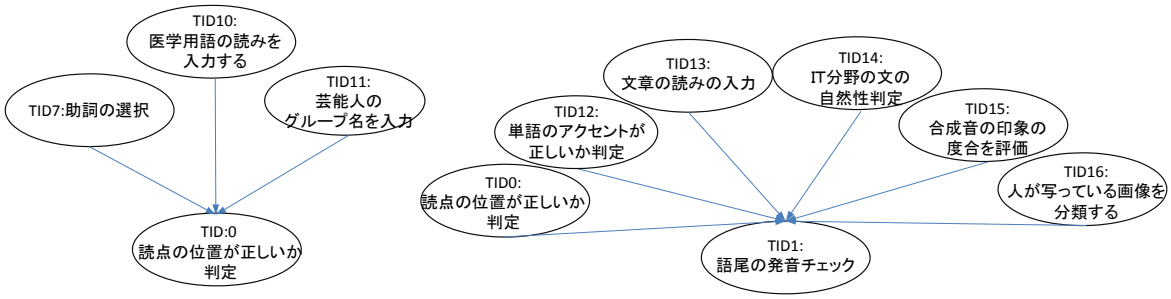
	0.アクセント含めた単語の読み方			1.画像の書き起こし(3)			10.文章中の一部分の読み方に違和感がないか判定(112)		
UID	精度	正解数	不正解数	精度	正解数	不正解数	精度	正解数	不正解数
2481	0	0	0	0	0	0	0	0	0
829	0	0	0	0	0	0	0	0	0
2469	0	0	0	97.1	272	8	0	0	0
2960	0	0	0	0	0	0	0	0	0
2090	0	0	0	0	0	0	0	0	0
1456	0	0	0	0	0	0	0	0	0
1179	0	0	0	0	0	0	92.8	841	65
1521	0	0	0	98.9	1005	11	0	0	0
1366	0	0	0	0	0	0	0	0	0
1922	0	0	0	99.6	1550	7	0	0	0
404	0	0	0	60.5	49	32	88.2	299	40
1862	94.9	1255	68	99.7	1924	6	97.7	21929	519
1897	0	0	0	0	0	0	0	0	0
516	0	0	0	98.5	199	3	0	0	0
1493	0	0	0	97.7	546	13	0	0	0
121	100	7	0	91.2	176	17	94.9	22627	1222

6.3 節で得られた 138 個のタスクカテゴリのうち，精度向上させたいタスクカテゴリとして平均精度が低い順に表 6.4 に示す．結果として，表 6.4 における精度改善対象となるタスクカテゴリ全てに対して有向グラフを得ることができた．その有向グラフの一例を図 6.5 に示す．この図は「Task category ID(TID)0: 読点の位置が正しいか判定」と「TID1: 語尾の発音チェック」，それぞれのタスクカテゴリにおける有向グラフである．

表 6.4: 精度改善タスクカテゴリ一覧

TID	タスクカテゴリ名	平均精度（%）
0	読点の位置が正しいか判定	73.8
1	語尾の発音チェック	82.6
2	対話パターン作成	83.4
3	有名人，芸能人の読み仮名を入力する	85.5
4	キーワードを分類	85.8
5	漢字の読み方の正誤判定	86.2
6	人名の音程の高低を入力する	87.5
7	助詞の選択	87.8
8	英単語の読みを入力する	88.7
9	単語の品詞を選択する	88.9

図 6.5 の左の有向グラフは精度改善対象となるタスクカテゴリ A を「TID0: 読点の位置が正しいか判定（図 6.6）」として式 (6.3) を適用した場合，タスクカテゴリ TID0 に関連性があるタスクカテゴリ X はそれぞれ「TID7: 助詞の選択」「TID10: 医学用語の読みを入力する」「TID11: 芸能人のグループ名を入力」（図 6.7）であることを示している．



TID7	TID10	TID11	P(TID0 | TID7,TID10,TID11) >= 90	P(TID0 | TID7,TID10,TID11)< 90
T	T	T	0.75	0.25

TID0	TID12	TID13	TID14	TID15	TID16	P(TID1 | TID0,TID12, TID13,TID14, TID15,TID16) >= 90	P(TID1 | TID0,TID12, TID13,TID14, TID15,TID16) <90
T	T	T	T	T	T	0.528	0.472

図 6.5: 精度改善対象タスクカテゴリにおける有向グラフの例

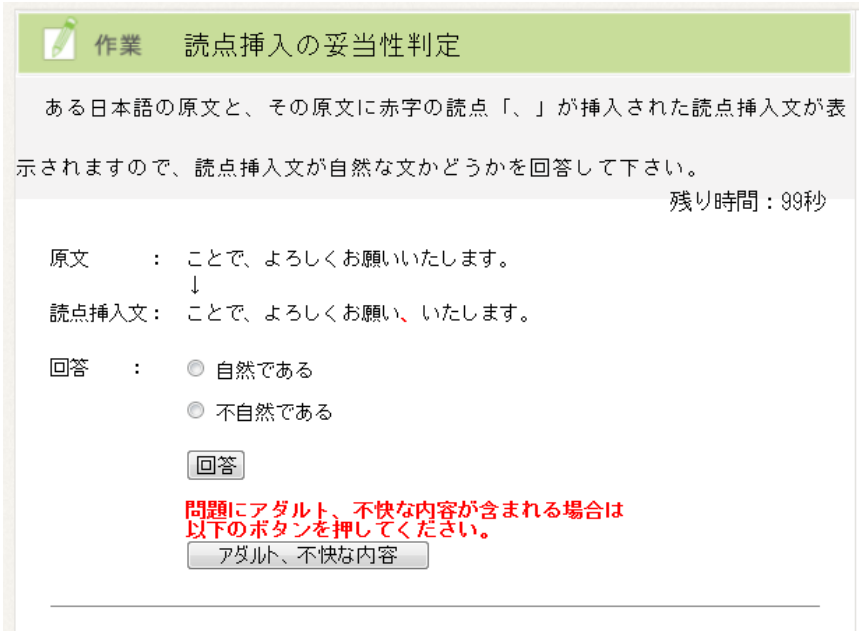
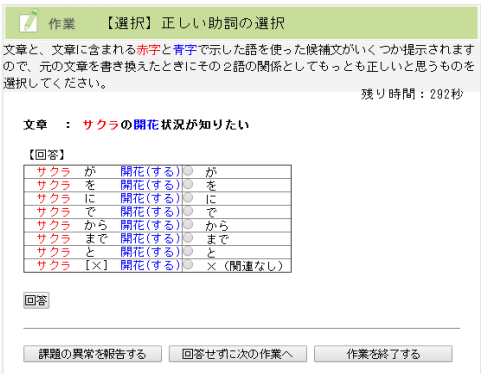
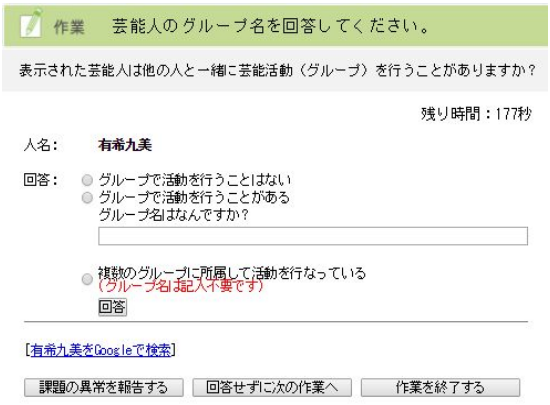


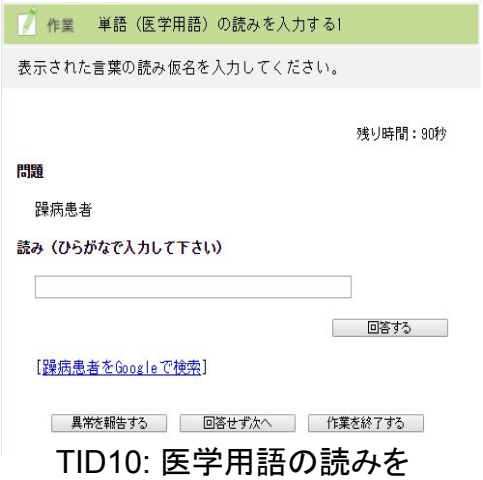
図 6.6: TID0: 読点の位置が正しいか判定



TID7: 助詞の選択



TID11: 芸能人のグループ名を入力



TID10: 医学用語の読みを

図 6.7: TID0 に対する学習タスク

ベイジアンネットワークを用いた場合は得られる有向グラフは複数層であるが，本研究では精度改善対象となるタスクカテゴリに直接影響を及ぼしているタスクカテゴリとの関係性に関してのみ述べる．ここで得られた直接影響を及ぼしているタスクカテゴリを学習タスクカテゴリとして扱う．また，比較対象として，ベイジアンネットワーク以外に決定木を用いてグラフを作成し，同様の実験を行った．決定木は最も影響のある要素を見つけ出しデータを分割していく手法である．決定木を用いて学習タスクカテゴリを作成するにあたって，PCSS は最初に精度向上対象タスクカテゴリ A における決定木を作成する．作成された決定木の例を図 6.8 に示す．決定木アルゴリズムは対象となるタスクカテゴリの精度に影響のあるタスクカテゴリを分岐ノードとして扱う．図 6.8 ではタスクカテゴリ B の結果精度が 90%以上であった場合タスクカテゴリ A を高精度で処理できることを示している．すなわち図 6.8 のような決定木が得られた場合，タスクカテゴリ B はタスクカテゴリ A の学習タスクカテゴリとして扱うことができる．

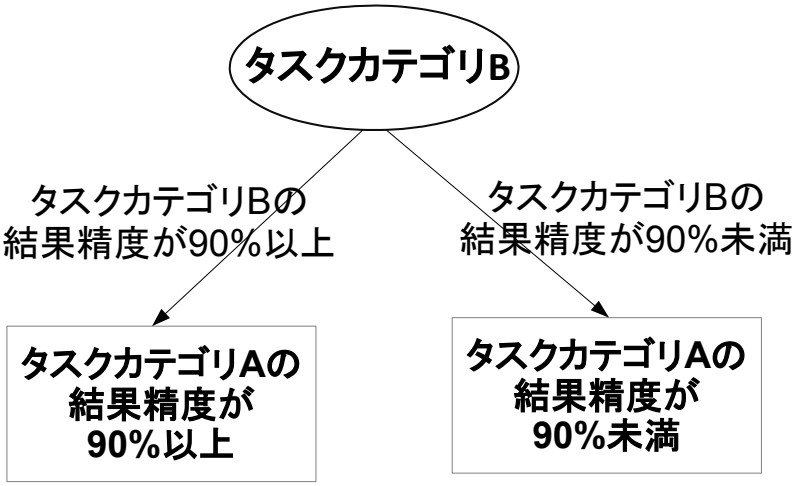


図 6.8: ベイジアンネットワークを決定木に用いた例

決定木を求めるためのアルゴリズムとして J48 を用いている．二分割法は用いず，枝刈りの閾値は 0.25，葉の最小数は 2 を用いた．階層の制限は行わなかった．

各有向グラフから得られた精度改善対象となるタスクカテゴリと，それぞれのタスクカ

テゴリに対するベイジアンネットワークから得られた学習タスクカテゴリと，決定木から得られた学習タスクカテゴリは表 6.5 のようになる．

表 6.5: 精度改善タスクカテゴリと対応する学習タスクカテゴリ

タスクカテゴリ名	ベイジアンネットワークから得られた学習タスクカテゴリ
TID 0:読点の位置が正しいか判定	TID 7:助詞の選択 TID 10:医学用語の読みを入力する TID 11:芸能人のグループ名を入力
TID 1:語尾の発音チェック	TID 0:読点の位置が正しいか判定 TID 12:単語のアクセントが正しいか判定 TID 13:文章の読みの入力 TID 14:ＩＴ分野の文の自然性判定 TID 15:合成音の印象の度合を評価 TID 16:人が写っている画像を分類する
TID 2:対話パターン作成	無し
TID 3:有名人，芸能人の読み仮名を入力する	TID 4:キーワードを分類 TID 11:芸能人のグループ名を入力 TID 14 ＩＴ分野の文の自然性判定
TID 4:キーワードを分類	TID 17:熟語のアクセントが正しいか判定
TID 5:漢字の読み方の正誤判定	TID 10:医学用語の読みを入力する TID 15:合成音の印象の度合を評価 TID 18:正しい文節の切れ目を選択 TID 19:Wikipedia の単語の読みを入力する
TID 6:人名の音程の高低を入力する	TID 3:有名人，芸能人の読み仮名を入力する TID 7:助詞の選択 TID 11:芸能人のグループ名を入力 TID 14:ＩＴ分野の文の自然性判定 TID 18:正しい文節の切れ目を選択 TID 20:文の言い換え文作成
TID 7:助詞の選択	TID 17:熟語のアクセントが正しいか判定 TID 21:言葉の共通語アクセントを選ぶ
TID 8:英単語の読みを入力する	TID 7:助詞の選択 TID 10:医学用語の読みを入力する
TID 9:単語の品詞を選択する	TID 2:対話パターン作成 TID 10:医学用語の読みを入力する TID 17:熟語のアクセントが正しいか判定 TID 22:文章の自然性判定

タスクカテゴリ名	決定木から得られた学習タスクカテゴリ
TID 0:読点の位置が正しいか判定	TID 24:人名の「苗字」と「名前」を区切る
TID 1:語尾の発音チェック	TID 11:芸能人のグループ名を入力
TID 2:対話パターン作成	TID 9:単語の品詞を選択する TID 25:読みがなと発音を組み合わせる
TID 3:有名人，芸能人の読み仮名を入力する	決定木作成に失敗
TID 4:キーワードを分類	TID 5:漢字の読み方の正誤判定 TID 26:単語の読みを入力
TID 5:漢字の読み方の正誤判定	TID 2:対話パターン作成 TID 21:言葉の共通語アクセントを選ぶ TID 27:単語の読み方の正誤判定
TID 6:人名の音程の高低を入力する	TID 21:言葉の共通語アクセントを選ぶ
TID 7:助詞の選択	TID 26:単語の読みを入力
TID 8:英単語の読みを入力する	TID 26:単語の読みを入力
TID 9:単語の品詞を選択する	TID 12:単語のアクセントが正しいか判定 TID 26:単語の読みを入力

第7章　ワーカーの段階的学習による精度向上手法の評価及び考察

本章では6章で述べた段階的学習手法の効果を確認するための実験とその結果から段階的学習手法の有効性に関して考察する．

7.1　学習の有無による各ワーカーの精度向上の実験

6章で得られたタスクカテゴリの相関関係の有効性を確認するために以下の様な実験を行った（図7.1）．実験対象となるタスクカテゴリは実験コストの問題から，精度向上対象タスクカテゴリの上位5つである，「TID0:読点の位置が正しいか判定」，「TID1:語尾の発音チェック」，「TID2:対話パターン作成」「TID3:有名人，芸能人の読み仮名を入力する」「TID4:キーワードを分類」を対象とした．

1. 初期状態の確認のために対象タスクカテゴリのタスクを50問処理させて精度チェックを行う．PCSSでは該当カテゴリのタスクにおける精度が60%以下になったワーカーはその作業をさせないという対応をしている．これは70%以下のワーカーを排除対象とすると悪意のないワーカーを多く排除してしまい，50%以下のワーカーを対象とすると，悪意のあるワーカーを排除しきれなかったというシステム運用上の経験からの数値である．また，90%以上のワーカーは既に優秀であるため学習の必要がないと判断し，この初期状態のチェックで正解率が60%以上90%以下のワーカーを以降の実験の対象とする．

2. (1)の条件を満たしたワーカーを以下の3グループに分け，それぞれ別の作業を行わせる．

(a) （ワーカーグループ1）
有向グラフから得られた学習タスクカテゴリのタスクそれぞれを10問ずつ実施させる．

(b) （ワーカーグループ2）
学習タスクカテゴリではなく，精度向上対象タスクカテゴリから「10 × ワーカーグループ1の学習タスクカテゴリの数」問作業させる．

(c) （ワーカーグループ3）
学習タスクカテゴリではなく，全く関係のないタスクカテゴリから「10 × ワーカーグループ1の学習タスクカテゴリの数」問作業させる．

3. (2) を実施後，(2) を処理した全員のワーカーにもう一度 (1) と同じタスクカテゴリの問題を50問処理させて精度改善効果をチェックする

4. (2)，(3) を3回繰り返す．

5. 一番最初に行った (1) の結果精度と一番最後に行った (3) の結果精度を比較して改善効果を測定する．

7.2　学習の有無による各ワーカーの精度向上結果の評価

この実験によって得られた効果を表7.1，表7.2に示す．各ワーカーグループには最初に40人の固定人数を割り当て，実際に作業した人数をテスト実施人数とし，実際に作業した人数のうち，正解率が60%以上90%以下のワーカーの人数を対象人数としている．また，ベイジアンネットワーク，決定木それぞれにおけるワーカーグループ1，2，3は比較のために同じタイミングで他の作業をさせずに実験しているが，ベイジアンネットワークの実験と決定木の実験には時間的にずれがある．つまり，ベイジアンネットワークのワーカーグループ2と決定木のワーカーグループ2、ベイジアンネットワークのワーカーグループ3と決定木のワーカーグループ3は同時には作業していない．これはシステム運用上，他のタスクも存在しており，全てのワーカーを実験に注力することが出来ないという点に起因している．その為，ワーカーグループ2,3はベイジアンネットワーク，決定木それぞれ

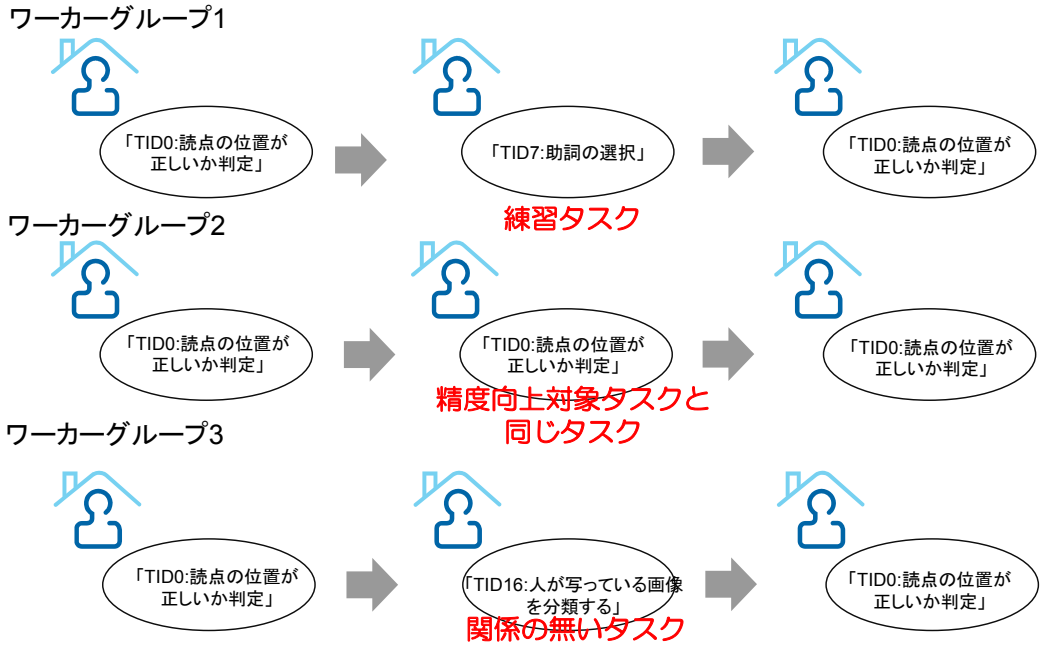


図 7.1: 段階的学習手法の効果を確認するための実験

に関して表に記載した．ベイジアンネットワークを用いて導出された学習タスクカテゴリを実施したワーカーグループ 1 は，対象タスクカテゴリ TID0 では 11.2 ポイント，対象タスクカテゴリ TID1 では 6.7 ポイント，対象タスクカテゴリ TID3 では 8.2 ポイント，対象タスクカテゴリ TID4 では 5.2 ポイントと，平均 7.8 ポイントの精度向上が確認できた．対象タスクカテゴリ TID2 に関しては有向グラフは得ることができたが，精度向上対象タスクカテゴリに対して影響を及ぼしているタスクカテゴリ（学習タスクカテゴリ）が存在しなかった．これらの結果から，ベイジアンネットワークの学習から導出された学習タスクカテゴリを実施することによって，精度向上対象タスクカテゴリの処理結果精度が大きく向上していることが確認できる．

また，学習タスクカテゴリを実施せずに同じタスクカテゴリを実施したワーカーグループ 2 はわずかながらの改善が見られるが，これは同一のタスクカテゴリ処理を継続することで作業に慣れた結果であると推測している．

また，比較のために決定木を用いて得られた学習タスクカテゴリを実施したワーカーグループ 1 の精度向上効果は平均 −0.6 ポイント，同じタスクカテゴリを実施したワーカーグ

ループ 2 の精度向上効果は平均 1.6 ポイント，関係のないタスクカテゴリを実施したワーカーグループ 3 の精度向上効果は平均 0.6 ポイントとあまり改善効果は見られなかった．また，TID3 における決定木は得られなかった．これらの結果から決定木で導出された学習タスクカテゴリをワーカーが実施しても精度改善効果は大きくなく，同一タスクカテゴリの実施，関係のないタスクカテゴリを実施した場合の作業の慣れによる精度向上とほぼ変わらないことが確認できた．

また，得られた表 7.1 のベイジアンネットワークの結果から学習タスクを行って結果精度が向上した人数の合計:23 人（表 7.1 におけるワーカーグループ 1 の「精度向上人数」の合計），学習タスクを行ったが結果精度が向上しなかった人数の合計:5 人（表 7.1 におけるワーカーグループ 1 の「対象人数」–「精度向上人数」の合計），学習タスクを行わず結果精度が向上した人数の合計:19 人（表 7.1 におけるワーカーグループ 2，3 の「精度向上人数」の合計），学習タスクを行わず結果精度も向上しなかった人数の合計:24 人（表 7.1 におけるワーカーグループ 2，3 の「対象人数」–「精度向上人数」の合計）を用いて 2×2 のテーブルを作成し，カイ二乗検定を行ったところ得られた P 値は 0.0014 であった．この検証結果により有意性があると判断することができる．

表 7.1: 学習タスクカテゴリ実施の有無によるタスク改善効果（ベイジアンネットワーク）

		ベイジアンネットワーク			
対象タスクカテゴリ	テストタイプ	テスト実施人数	対象人数	精度向上人数	平均精度向上値（point）
TID 0:読点の位置が正しいか判定	練習タスクカテゴリ（ワーカーグループ 1）	7	3	3	11.2
	同一タスクカテゴリ（ワーカーグループ 2）	8	3	2	3.4
	関係ないタスクカテゴリ（ワーカーグループ 3）	7	4	2	3.5
TID 1:語尾の発音チェック	練習タスクカテゴリ（ワーカーグループ 1）	24	8	6	6.7
	同一タスクカテゴリ（ワーカーグループ 2）	13	4	1	-3.0
	関係ないタスクカテゴリ（ワーカーグループ 3）	13	7	4	1.6
TID 2:対話パターン作成	練習タスクカテゴリ（ワーカーグループ 1）	学習タスクカテゴリ無し			
	同一タスクカテゴリ（ワーカーグループ 2）	学習タスクカテゴリ無し			
	関係ないタスクカテゴリ（ワーカーグループ 3）	学習タスクカテゴリ無し			
TID 3:有名人の読み仮名を入力する	練習タスクカテゴリ（ワーカーグループ 1）	8	7	6	8.2
	同一タスクカテゴリ（ワーカーグループ 2）	8	7	3	-0.5
	関係ないタスクカテゴリ（ワーカーグループ 3）	8	6	1	-0.3
TID 4:キーワードを分類	練習タスクカテゴリ（ワーカーグループ 1）	12	10	8	5.2
	同一タスクカテゴリ（ワーカーグループ 2）	7	6	4	3.3
	関係ないタスクカテゴリ（ワーカーグループ 3）	8	6	2	0.6

表 7.2: 学習タスクカテゴリ実施の有無によるタスク改善効果（決定木）

		決定木			
対象タスクカテゴリ	テストタイプ	テスト実施人数	対象人数	精度向上人数	平均精度向上値（point）
TID 0:読点の位置が正しいか判定	練習タスクカテゴリ（ワーカーグループ 1）	8	8	4	-3.2
	同一タスクカテゴリ（ワーカーグループ 2）	31	17	6	1.3
	関係ないタスクカテゴリ（ワーカーグループ 3）	6	0	0	0
TID 1:語尾の発音チェック	練習タスクカテゴリ（ワーカーグループ 1）	12	5	2	-1.0
	同一タスクカテゴリ（ワーカーグループ 2）	13	6	3	1.1
	関係ないタスクカテゴリ（ワーカーグループ 3）	11	5	3	1.2
TID 2:対話パターン作成	練習タスクカテゴリ（ワーカーグループ 1）	13	7	3	-0.5
	同一タスクカテゴリ（ワーカーグループ 2）	10	9	5	1.7
	関係ないタスクカテゴリ（ワーカーグループ 3）	13	10	6	0.8
TID 3:有名人の読み仮名を入力する	練習タスクカテゴリ（ワーカーグループ 1）	決定木作成できず			
	同一タスクカテゴリ（ワーカーグループ 2）	決定木作成できず			
	関係ないタスクカテゴリ（ワーカーグループ 3）	決定木作成できず			
TID 4:キーワードを分類	練習タスクカテゴリ（ワーカーグループ 1）	14	8	4	2.4
	同一タスクカテゴリ（ワーカーグループ 2）	12	6	3	2.3
	関係ないタスクカテゴリ（ワーカーグループ 3）	11	3	2	0.6

7.3 学習の有無による各ワーカーの精度向上結果の考察

実験結果から，マイクロタスク型クラウドソーシングにおけるワーカーの品質改善を行うにあたって，ベイジアンネットワークを用いてワーカーの行動履歴から学習タスクを導出する段階的学習手法は有効であるが決定木を用いた手法はあまり効果が得られないことがわかった.

タスクカテゴリ A を解析するにあたって，タスクカテゴリ X_0 におけるワーカーの処理結果 $T(X_0)$ が高精度である確率 $P(T(X_0))$ を目的変数とし，X_0 以外のタスクカテゴリ $X_1, X_2, X_3 \cdots X_n$ におけるワーカーの処理結果 $T(X_1)$，$T(X_2)$，$T(X_3) \cdots T(X_n)$ が高精度である確率 $P(T(X_1), T(X_2), T(X_3) \cdots T(X_n))$ を説明変数としている．決定木を用いた解析では，決定木の各ノードには分類する属性が対応付けられ，ノードを結ぶリンクには属性値が対応付けられる. 決定木を用いてタスクカテゴリ X_0 を解析するにあたって，属性を X_0 以外のタスクカテゴリ $X_1, X_2, X_3 \cdots X_n$, 属性値を処理結果が高精度であるか否かとして決定木を作成した. 決定木は影響の大きい要素を優先的に選択して作成されており，本研究ではこの優先的に選択されている属性（タスク）を学習タスクとして用いている（図 6.8）. このように決定木の解析では説明変数と目的変数の関係だけを考えて解析を行っているが，実際のタスクカテゴリ同士は 6 章で述べたように，タスクカテゴリ X_i がタスクカテゴリ X_j の学習タスクカテゴリとなりうる可能性があるため，タスクカテゴリ X_i とタスクカテゴリ X_j は相互的に影響がないとは言いがたい．説明変数と目的変数の関係だけを考える決定木に対して，ベイジアンネットワークでは説明変数間の関係を学習しているため [Okamoto 08],「タスクカテゴリ X_i を高精度で処理することができたのでタスクカテゴリ X_j を高精度で処理することができた」という因果関係を含んだ $P(T(X_i) \mid T(X_j))$ を学習することが出来ており，決定木よりも精度向上効果のある学習タスクを算出できたものと推測している.

本実験における意味のある因果効果とは，あるワーカーに今までと違う処理条件 (=学習タスク) を与えることで，そのワーカーの反応に望ましい変化が現れるという現象である. この因果効果を測定するために，学習タスク X_j を実施させたワーカーグループでのタスク X_i における効果 Y_j と学習タスク以外のタスク X_k を実施させたワーカーグループのタスク X_i における効果 Y_k とするとそれぞれの差，つまり学習タスクを実施することによる効果の期待値 E

$$E[Y_j - Y_k] = E[Y_j] - E[Y_k] \tag{7.1}$$

を集団での因果効果と定める事ができる [宮川 04]．さらにワーカーに学習タスクを実施させるか (学習タスク X_j を実施) させないか (学習タスク以外のタスク X_k を実施) を示す変

数を考え，これを確率変数 W として定式化する．この W も 2 値変数で，$W = 1$ は学習タスク X_j を実施を，$W = 2$ は学習タスク以外のタスク X_k の実施を意味する．すると実験によって得られた結果である 7.1 は $W = 1$ のワーカーにおける精度向上効果と Y_j と $W = 2$ のワーカーにおける精度向上効果 Y_k となる．よって

$$E[Y_j \mid W = 1] - E[Y_k \mid W = 2] \tag{7.2}$$

は計算することが可能となる．一般的には式 (7.1) と式 (7.2) は異なるが，本実験ではワーカーに学習タスクを実施させるか $(W = 1)$ させないか $(W = 2)$ は無作為に割りつけを行っているため，W と（Y_j，Y_k）は統計的に独立になる．このとき

$$E[Y_j \mid W = 1] = E[Y_j \mid W = 2] = E[Y_j] \tag{7.3}$$

$$E[Y_k \mid W = 1] = E[Y_k \mid W = 2] = E[Y_k] \tag{7.4}$$

が成り立つため [宮川 04]，式 (7.2) の条件付き期待値と式 (7.1) の期待値は等しくなる．このように，本実験は無作為実験であるため，集団の因果的効果を偏り無く推定できていると考えることができる．

さらに，決定木は影響の大きい要素を優先して解析していくため，どの順で解析したがが決定木の作成に大きく影響する．そのため，対象となるデータに外れ値や偏りが多く存在していた場合は優先度に影響を与えてしまう可能性が大きい．決定木の有効性が低い理由として，今回の解析対象となるクラウドソーシングのタスク処理はワーカーが不特定多数であり，品質が一定していないデータであるため，それらのデータも精度改善効果が得られない一因であると推測している．

本実験では精度改善対象タスクカテゴリに対して有向グラフ上で直接影響を与えていると解析されたタスクカテゴリのみを学習タスクカテゴリとして用いている．例えば図 6.4 では精度改善対象をタスクカテゴリ D とした場合，タスクカテゴリ B とタスクカテゴリ C のみを学習タスクカテゴリとし，タスクカテゴリ A は学習タスクカテゴリとして扱っていない．これはタスクカテゴリ A はタスクカテゴリ B と C と比較してタスクカテゴリ D に

対する影響力が少ないため計算量および実験コストを削減するために省略した．全ての影響あるタスクカテゴリを学習タスクカテゴリとして扱った場合の実験を低コストで行う方法は今後の課題である．

また，精度向上タスクが学習タスクとなってしまった場合は，有向グラフにおいて1階層の相互に影響し合うループが発生する．この場合は精度向上対象タスクカテゴリを繰り返し実施する，すなわち実験におけるワーカーグループ2のケースで精度が向上すると予測している．しかし，現時点ではこのようなケースは発生しておらず未検証であるため，検証は今後の課題である．

精度改善対象となるタスクカテゴリと，得られた学習タスクカテゴリの間には一見関連性がないように見えるものも存在する.しかし，タスクカテゴリの内容的に関連性が少なくても，ベースとなる知識やタスクデザインなどの点で共通する点があるものと推測される．

従来の教育実践及び教育システムにおいてはボトムアップ方式の教授方法が有効である．クラウドソーシング環境においてボトムアップ方式の教授方法を用いる場合，学習タスクは精度向上対象のタスクよりも簡単であることが理想的である．しかし，クラウドソーシングにおいてタスクAがタスクBより「簡単である」とは「タスクBを処理するために必要な知識よりも少ない知識でタスクAの処理が可能」と考えた場合，学校教育のようにタスクAを処理するために必要な知識の部分知識のみで処理が可能な学習タスクBを設計し，タスクBでワーカーを学習させることは非常に困難である．クラウドソーシングでは大量のタスクが存在し，また教師という熟達した管理者も存在しないため，学習タスクBを作成することがシステム管理者，リクエスタにとっては非常に高コストであるためである．本研究はシステム管理者及びリクエスタに負担をかけること無くワーカーに学習させるために「タスクAを処理するのに必要な知識の部分知識のみ」で構成された学習タスクBではなく「タスクAを処理するのに必要な知識の部分知識を含んだ」既存のタスクCで学習させることは出来ないかという仮説を立て，効果があることを実証している．

例えば，タスクAを処理するために必要な知識が知識aであり，タスクCを処理するための必要な知識が知識aの部分知識a’と知識cだった場合，タスクCを処理するにはタスクAよりも多くの知識が必要である場合がある．しかし，そのようなケースでもワーカーが知識cを既にもっていた場合はタスクCはタスクAの学習タスクとなることが出来る．

精度向上タスクカテゴリ「TID9 : 単語の品詞を選択する」と学習タスクカテゴリ「TID2 : 対話パターン作成」の場合，TID9 を処理するための必要な知識は「日本語文法の知識 (a)」だが，TID2 を処理するためには「一般的な日本語の知識 (a ’)」と「対話文章の作文能力 (c)」が必要になる．ワーカーによっては正確ではない文法で日本語会話を行っている可能性があり (「一般的な日本語の知識 (a ’)」と「対話文章の作文能力 (c)」の知識は持っているが「日本語文法の知識 (a)」の知識に乏しいケース)，その場合でも TID2 を処理する過程で様々な文章を作文していくうちに日本語の文法に慣れ親しんでいくことで TID9 の処理で必要な「日本語文法の知識 (a)」，すなわち「単語が人名であるかどうか」「単語が地名であるかどうか」「単語が名詞であるかどうか」などを学習していると考えている．

また，精度向上対象タスクカテゴリ「TID 4:キーワードを分類」と学習タスクカテゴリ「TID 17:熟語のアクセントが正しいか判定」の場合，TID4 を処理するために必要な知識は「日本語の単語に関する知識 (a)」だが，TID17 を処理するためには「一般的な日本語の知識 (a ’)」と「アクセントに関する知識 (c)」が必要になる．ワーカーによっては文法や単語を意識せずに一般的な発音で会話しているケースが存在する (「一般的な日本語の知識 (a ’)」と「アクセントに関する知識 (c)」は持っているが,「単語」の概念など「日本語の単語に関する知識 (a)」の知識に乏しいケース)．その場合でも TID17 で文章の中の熟語のアクセントを処理する過程で文章のどの部分が熟語となっているかなどをを確認し,「日本語の単語に関する知識 (a)」，すなわち「単語は文章のどこで切るのか」などを学習していると考えている．

同様に「TID 3:有名人，芸能人の読み仮名を入力する」において必要な知識は「日本語に読み仮名を入力する知識 (a1)」,「芸能人に関する知識 (a2)」,「最新の情報をチェックする能力 (a3)」と考えることが可能であり,「TID 4:キーワードを分類」で必要な知識は「日本語の単語に関する知識 (a1 ’)」,「TID 11:芸能人のグループ名を入力」で必要な知識は「芸能人に関する知識 (a2)」,「TID 14 : ＩＴ分野の文の自然性判」で必要な知識は「最新の情報をチェックする能力 (a3)」「ＩＴ関連の知識 (c3)」と考えることが可能である．この場合，精度向上タスクカテゴリ TID 3 に対して，学習タスクカテゴリ TID4 の部分知識 a1 ’ は TID3 で必要な部分知識 a1 に，学習タスクカテゴリ TID11 の部分知識 a2 は TID3 で必要な部分知識 a2 に，学習タスクカテゴリ TID14 の部分知識 a3 は TID3 で必要な部分知識

a3 に，それぞれ影響を及ぼしていると考えている．

さらに，ユーザーインタフェイスの設計から学習効果を考察する．精度向上対象タスクカテゴリ「TID 0:読点の位置が正しいか判定」における学習タスクカテゴリ「TID 7:助詞の選択」は選択式のユーザーインターフェイスで日本語文章の特定の点に関して回答するという点，精度向上対象タスクカテゴリ「TID 1:語尾の発音チェック」における学習タスクカテゴリ「TID 12:単語のアクセントが正しいか判定」は与えられた複数音声データを再生し，正しい音声を選択するというユーザーインターフェイスという点，精度向上タスクカテゴリ「TID 3:有名人の読み仮名を入力する」における学習タスクカテゴリ「TID 11:芸能人のグループ名を入力」は人名をテキストで入力するユーザーインターフェイスという点，精度向上タスクカテゴリ「TID 4:キーワードを分類」における学習タスクカテゴリ「TID 17:熟語のアクセントが正しいか判定」は選択式のユーザーインターフェイスという点でそれぞれ共通点を持つと考えられる．しかし，PCSS はマイクロタスク型のクラウドソーシングなので，ユーザーインタフェイスは非常に簡易なものが中心となっており，ユーザーインタフェイスを原因とした精度低下が発生しているとは考えにくい．そのため，学習タスクの導出にはユーザーインタフェイスなどの類似性ではなく，必要としている知識の類似性が重要であると考えている．

このようにクラウドソーシング環境では発見が困難な学習タスクを自動的に解析，抽出することが可能になったという点が，本研究の貢献点であると考えている．

また，ワーカーのモチベーション低下はタスクの処理結果精度の低下につながり，ワーカーのモチベーションは報酬やタスクの面白さに影響されることがわかっている [Kittur 08]．今回実施した実験では対象となるワーカーは割り当てられた学習タスクカテゴリ以外のタスクを行うことができなかったため，モチベーションの低下の原因となり，最終的に精度が低下するケースや，最後まで作業を完了させずに途中でやめてしまうケースにつながってしまった可能性がある．より正確に段階的学習手法の効果を確認するためにモチベーションの低下しないテスト方法の構築も今後の課題としたい．

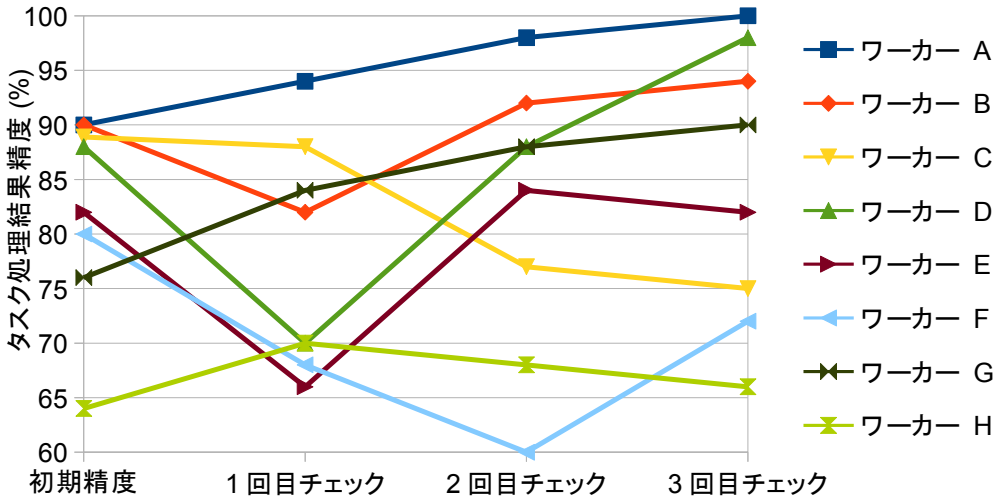


図 7.2: 精度改善対象タスクカテゴリ TID1 におけるワーカーの成長パターン

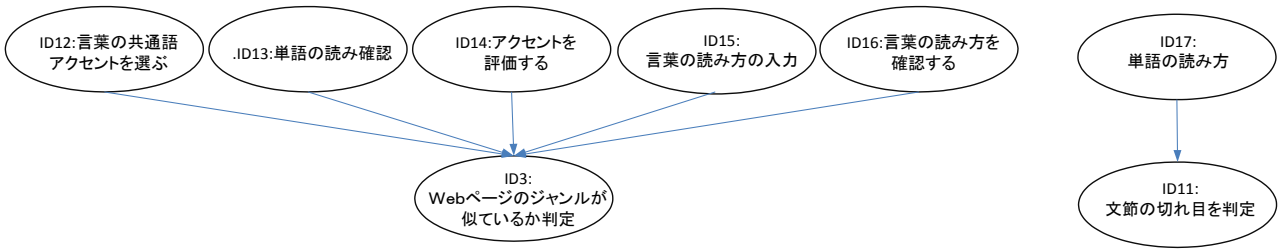
今回の実験で得られたワーカーの成長パターンは様々であった．TID1 におけるワーカーの成長パターンを図 7.2 に示す．ワーカーの成長パターンを，(1) 継続的に精度が向上していくパターン（ワーカー A，G），(2) 精度は上下するが最終的に向上するパターン（ワーカー B，D，E，H），(3) 精度は上下するが最終的に低下するパターン（ワーカー F），(4) 精度が継続的に低下するパターン（ワーカー C），の 4 パターンに分類した．今回の実験ではパターン (1) と (2) のワーカー数の和がパターン (3) と (4) のワーカー数の和よりも多いため，段階的学習手法は効果があると判断している．例えば TID1 においてはパターン (1) とパターン (2) のワーカー数の和は 6 人，パターン (3) と (4) のワーカー数の和は 2 人である．パターン (3) と (4) のような最終的に低下してしまうケースはワーカーの個人情報などが影響しているのではないかと推測しており，個人情報を加味した段階的学習手法の提案を検討したい．

今回の実験では 10 個の精度改善対象タスクカテゴリのうち 9 個の精度改善対象タスクカテゴリにおいて学習タスクカテゴリとして扱うことができるノードを持つ有向グラフを得ることができた．TID2 に関しては有向グラフを得ることができたが学習タスクカテゴリを得ることが出来ていない．これは精度改善対象タスクカテゴリが有向グラフのトップに

配置されてしまったためである．我々の以前の実験 [Ashikawa 14] では本実験の 1853 万タスクよりも少ない 700 万タスクを対象としたが，結果として得られた精度改善タスクカテゴリ（表 7.3）に対して図 7.3 に示される 2 つの有向グラフしか得ることができなかった．それ以外のタスクでは学習タスクカテゴリが得られない有向グラフしか得ることができなかった．この結果から，ワーカーの行動履歴の量は学習タスクの導出に影響を与えていることがわかる．

表 7.3: 小規模データにおける精度改善タスクカテゴリ一覧

L-TID（小規模データにおける TID）	タスクカテゴリ名	平均精度（%）
1	キーワード分類	85.8
2	品詞判定	86.6
3	Web ページのジャンルが似ているか判定	90.1
4	アクセントを評価する	90.4
5	単語の読み方確認	92.3
6	アクセントを含めた読み方確認	93.0
7	アクセント選択	93.2
8	言葉の読み方を判定	94.1
9	言葉の読み方の入力	94.4
10	自然文判定	94.7
11	文節の切れ目を判定	95.8



ID12	ID13	ID14	ID15	ID16	P(ID3 | ID12,ID13,ID14,ID15,ID16)
T	T	T	T	T	0.833
T	T	T	T	F	0.167

ID17	P(ID11 | ID17)
T	0.573
F	0.427

図 7.3: 小規模データから得られた有向グラフ

一方，小規模データから得られた学習タスクにおいても 7.1 節で実施した実験と同様の実験を行うことで表 7.4 に示すように改善効果を確認することが出来た．これによりワーカーの行動履歴の量に差があっても得られた学習タスクの有効性は変わらないことが予想される．

表 7.4: 小規模データから得られた学習タスク実施の有無によるタスク改善効果

対象タスクカテゴリ	学習タスクカテゴリ	対象人数	精度向上人数	平均精度向上値（point）
L-TID3: Ｗｅｂページのジャンルが似ているか判定	練習タスクカテゴリ（ワーカーグループ 1）	5	5	10.8
	同一タスクカテゴリ（ワーカーグループ 2）	6	3	2.2
	関係ないタスクカテゴリ（ワーカーグループ 3）	7	4	0.3
L-TID11: 文節の切れ目を判定	練習タスクカテゴリ（ワーカーグループ 1）	10	10	9.7
	同一タスクカテゴリ（ワーカーグループ 2）	10	5	2.9
	関係ないタスクカテゴリ（ワーカーグループ 3）	11	5	1.4

これまで述べてきたように，学習タスクはワーカーの行動履歴から導出される．しかし，ワーカーの行動履歴は常に増加し，さらに段階的学習手法によって変化していく．そのため効果的な学習タスクを導出するには適宜ワーカーの最新の行動履歴を解析しなくてはならないが，計算量が大きく非常に高コストである．ワーカーの成長に応じて効果的に段階的学習を行わせるためには，最適な解析スケジュールの設定が急務である．

第8章　ワーカーのフィルタリング及び段階的学習の事例紹介

PCSS を用いて知識処理研究に必要な語彙を収集した事例について述べる． 収集のフローを図 8.1 に示す．まず始めに，Web クローラを用いた大規模テキストの収集を行い，続いて収集したテキストから未知語の候補を自動抽出する．そして最後に，PCSS を用いて未知語候補から単語として適当なものだけを絞り込み，知識処理研究の一環である音声認識や音声合成の辞書を構築するために必要な品詞や読み仮名，アクセント等の単語情報を付与する．

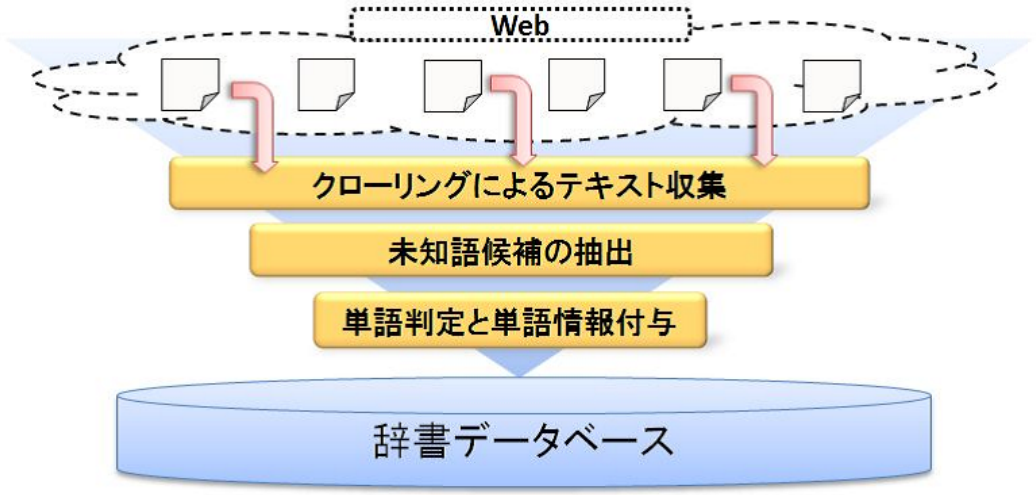


図 8.1: 語彙抽出フロー

8.1　語彙の重要性

知識処理研究では基礎となるデータとして大量の語彙情報を必要とする．例えば現在の形態素解析では辞書を用いるが，その辞書で語彙が不足していた場合，未知語が多く発生してしまいうまく形態素解析を行うことが出来ず，望む結果が得られないことがある．例えば形態素解析を用いる研究の例として電子書籍読み上げが挙げられるが，図 8.2 よりわかるように未知語による読み誤りは精度を低下させる大きな原因となっている．

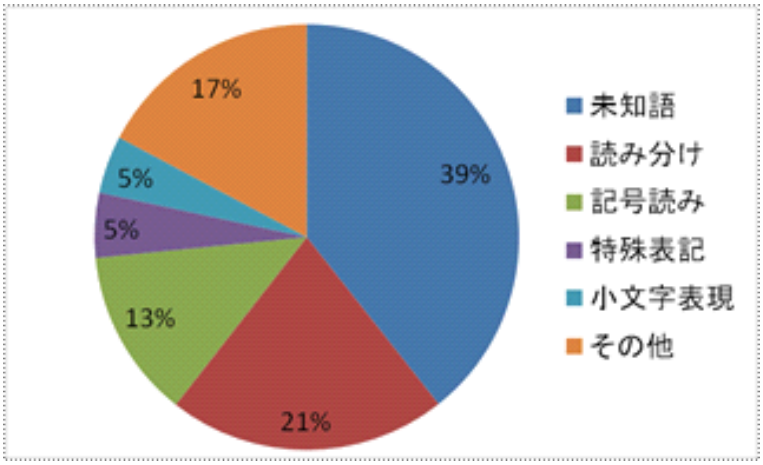


図 8.2: 電子書籍読み上げにおける読み誤り原因

また，新語は常に発生しており，それらの新語を語彙として常に辞書に登録する必要がある．その為には Web などにおける最新の大量のテキストデータから語彙を抽出し，音声処理や自然言語処理と言った知識処理研究に必要な情報である「読み仮名」「アクセント」「品詞」といった情報を付加していかなければならない．

本研究では図 8.3 に示すフローで Web から新語の抽出を行っている

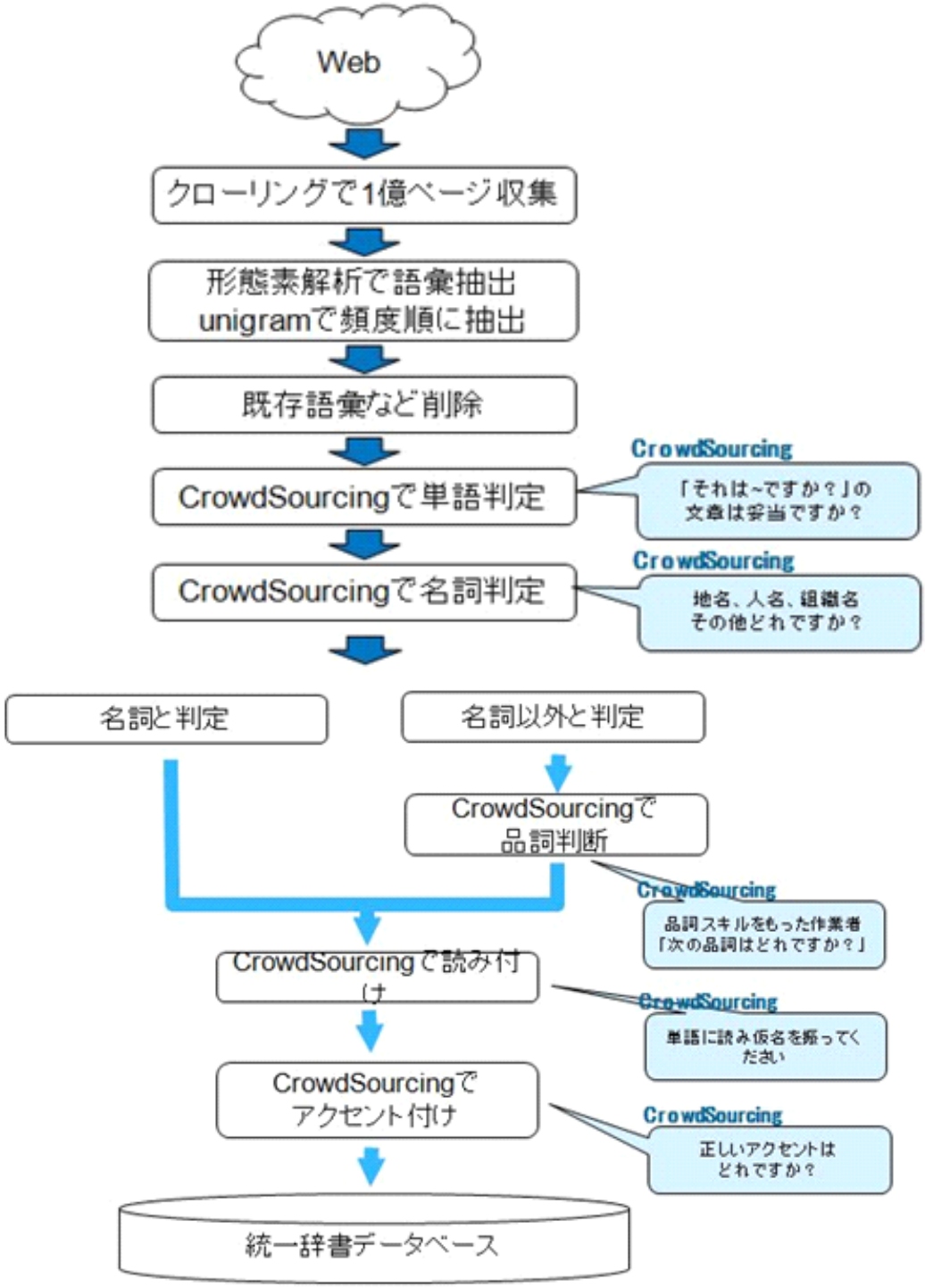


図 8.3: Web からの新語抽出フロー

8.2　クローリングによるテキスト収集

Web テキストには，固有名詞や新語などの未知語が頻繁に出現する．こうした未知語を獲得するコーパスとして，Web テキストを収集する．本研究では，OpenDirectory[1] の URL をシードとして，Apache Nutch[2] を用いて収集した．

獲得したテキストの情報を表 8.1 に示す．5.2 億ページから日本語文 125 億文を得ることが出来た．

表 8.1: 獲得した Web テキスト

獲得ページ数	517,239,154
日本語ページ数	319,570,805
文数	12,504,868,218

8.3　未知語候補の抽出

8.2 節で収集したテキストから未知語の候補を抽出する．抽出処理は以下のステップで行った．

1. テキストに対して点予測手法 [森 11] による単語分割を実施
2. 単語分割結果から辞書未登録文字列を取得
3. 単語分割結果を用いて単語 Ngram を作成
4. 単語 Ngram を用いて辞書未登録文字列の中から未知語候補を選出

この一連の処理によって 125 億文のテキストから 23 万語の未知語候補を抽出することができた．

[1] http://www.dmoz.org/World/Japanese/
[2] http://nutch.apache.org/

8.4 単語判定と単語情報付与

8.3 節の方法で作成された未知語候補には，単語として適当でないものが残っている可能性が高い．また，抽出した単語に対して音声処理に必要な情報を付与しなくてはならない．これらの情報収集を PCSS の以下の 4 タスクとして行った．

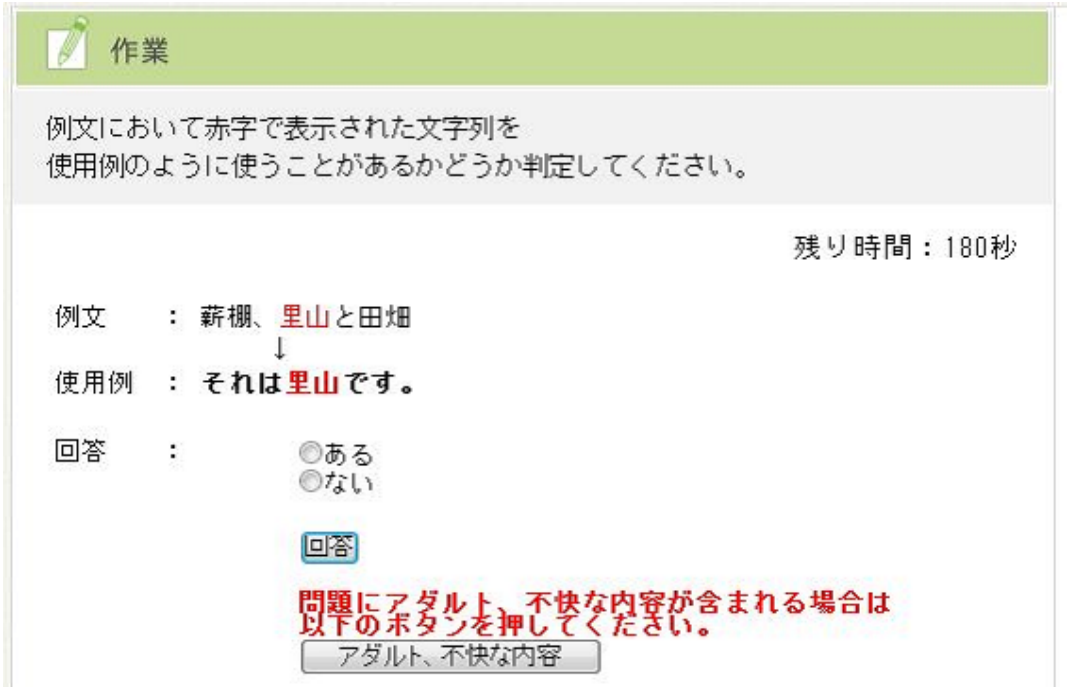


図 8.4: 単語判定タスク

1. 単語判定タスク

   タスクデザインを図 8.4 に示す．このタスクではワーカーに対して 8.3 節の方法で作成された未知語候補を「それは（未知語候補）です」という問題文に加工して表示し，「問題文は日本語して自然か否か」という選択をさせた．「日本語として自然である」と回答された場合，その文章に含まれる未知語候補を未知語として扱う．例えば図の例では抽出された語彙「里山」を単語かどうか判定するために「それは里山です」という例文を用いた．「里山」は単語として判断されるのが理想であるため，この文章は問題あると回答されるのが望ましい．しかし，形態素解析の結果によっては「お子ちゃまと一緒に」という文から「ちゃま」という単語が未知語として抽出され

てしまう場合がある．この場合は「それはちゃまです」という文が例文として提示される.「ちゃま」は単語として判断されないのが理想であるため,「ない」という結果が得られるのが望ましい．

2. 品詞付与タスク

タスクデザインを図8.5に示す．このタスクでは名詞とそれ以外の品詞に分ける作業を行なっている．名詞に関しては「人名」「地名」「組織名」「その他の名詞」に再分類している．(1) で単語として適切であると判定された未知語に単語抽出元の前後の文章を付与して問題文に加工して表示し,「人名」「地名」「組織名」「その他の名詞」「名詞以外」を選択させた．

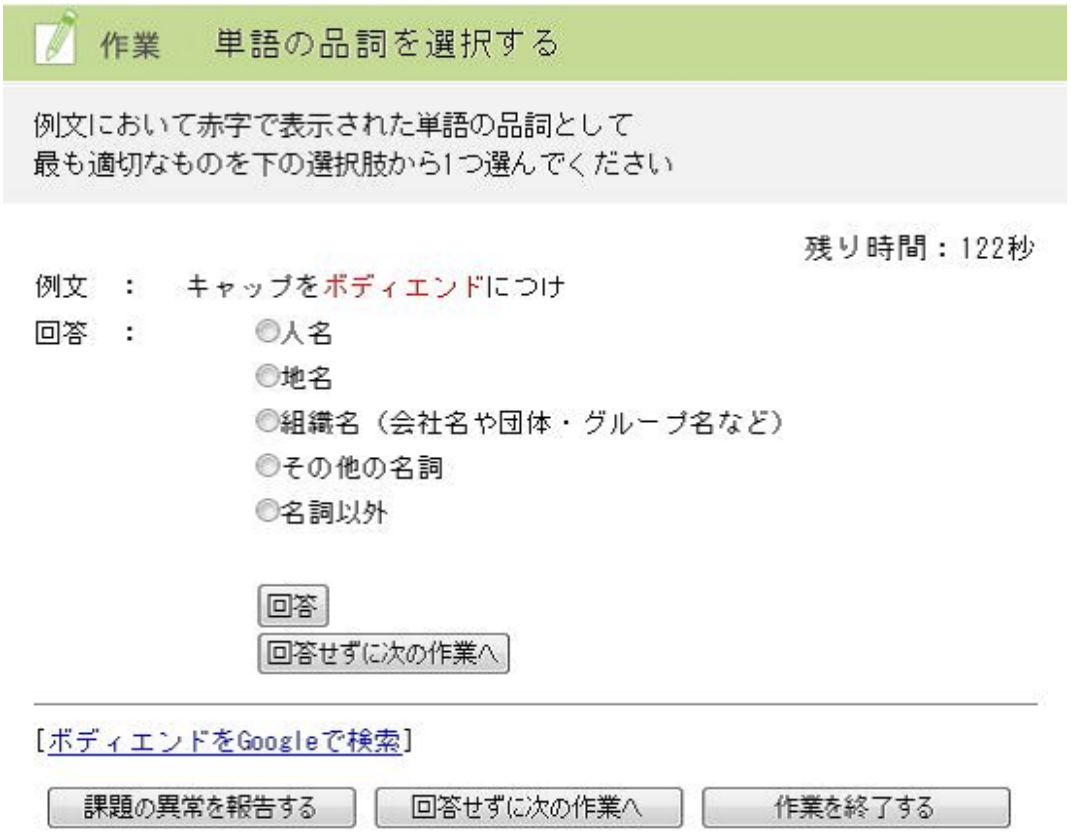


図 8.5: 品詞付与タスク

3. 読み付与タスク

タスクデザインを図 8.6 に示す．このタスクでは (2) で名詞と判定された未知語を問題として表示し，その読みを入力させ，その結果を未知語に対する読みと判定した．最初は単語抽出を行わず文章への読みつけ作業を行った．一つの文章を 3 人に対して出題し，3 人，もしくは 2 人が一致したものを正解とした．しかし，この結果としては 3 人一致が 17.8%，2 人一致が 38.5%，不一致が 37.3%とずれが大きい結果となった．これは長文入力において入力ミスなどの誤差が多く，長文のためミスの影響範囲が大きいなどが原因であると判断し，図 8.6 のように単語への読みつけを行う方式へと変更した．これにより精度が大幅に向上した．また，単語にしたということで一作業あたりの報酬を下げることで全体のコストを上げることなく精度改善を可能とした．

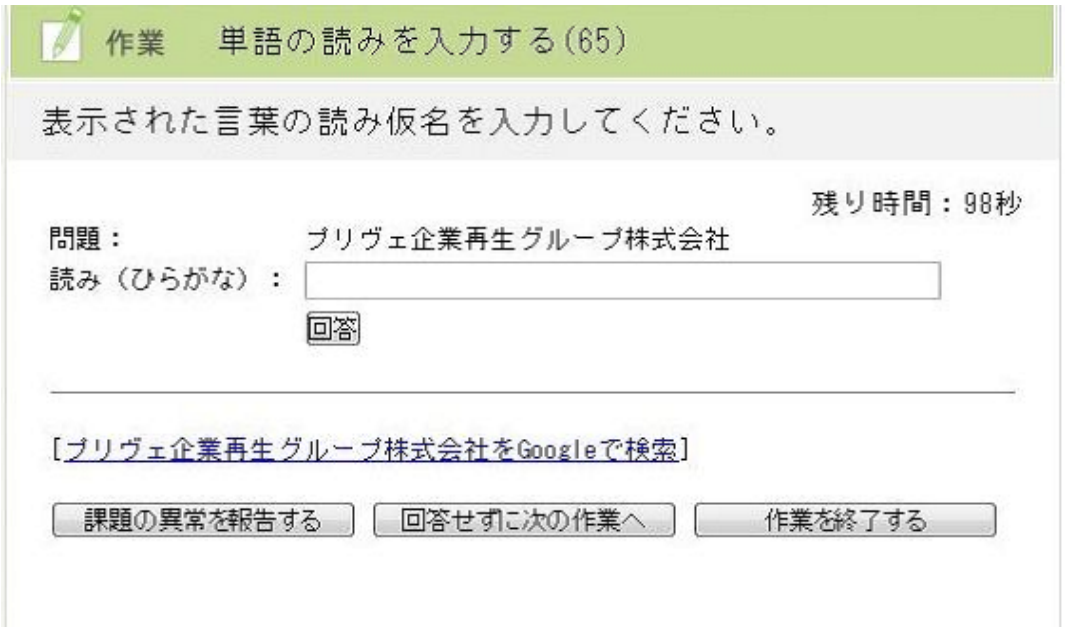


図 8.6: 読み付与タスク

4. アクセント付与タスク

タスクデザインを図 8.7 に示す．このタスクでは (3) で付けられた読みから推定されるアクセント候補から合成した音声を用い，どれが自然かを選択させた．その結果を未知語に対するアクセントと判定した．この作業は難易度が高いため，アクセントスキル保持ワーカー 163 名にのみタスク処理させている．

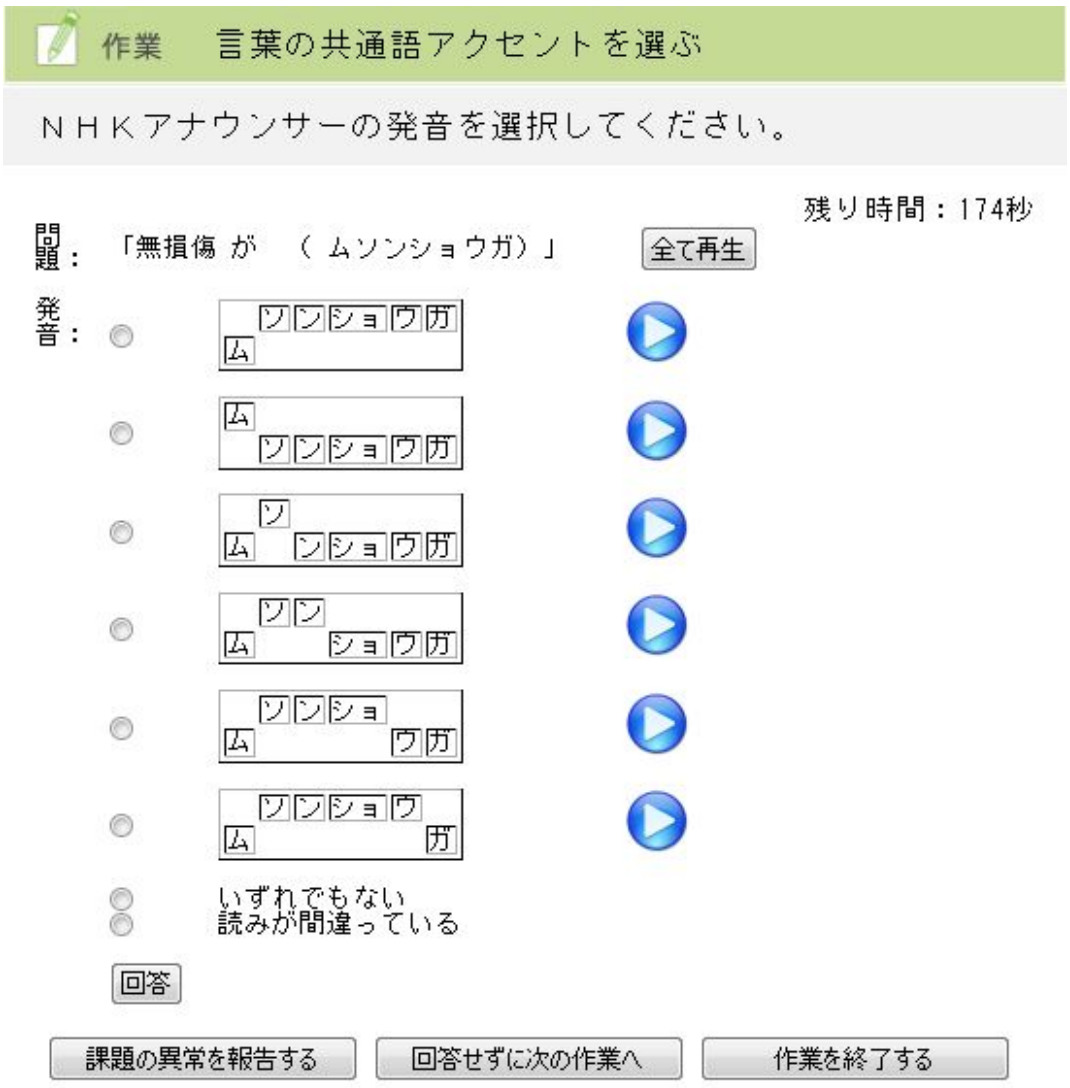


図 8.7: アクセント付与タスク

各タスクは 3 人に出題され，2 人以上一致した回答を有効なデータとして扱う．ただし，(1) の単語判定タスクは高精度であることを求められるため，3 人が一致した回答のみを有効なデータとして扱った．また，ワーカーが設問が不適切であると判断した場合は「パス」を選択できるようにしている．通常のパスであれば回答権は他のワーカーに移動するが，6 回以上パスが行われた場合はその問題は不適切と判定されて排除される．PCSS ではリクエスタからの中断依頼がない限り，出題した全ての問題に対して回答かパスの処理が行われるまで出題される．各カテゴリにおけるタスク処理結果から無作為に 10000 件の結果

データを抽出し，一致率を調査した結果を表 8.2 に示す．

表 8.2: 各タスクの作業結果における一致率

	3 人一致	2 人一致	不一致	不適切
単語判定カテゴリ（2 択）	71.1%	28.9%	0.0%	0.0%
読み付けカテゴリ	75.6%	16.7%	2.4%	5.3%
品詞カテゴリ	84.3%	2.4%	13.2%	0.1%
アクセントカテゴリ	66.0%	28.5%	3.9%	1.6%

8.5 結果

以上の処理を用いて Web から得られた語彙数を表 8.3 に示す．125 億文の Web テキストから 14 万語の語彙を獲得することが出来た．獲得できた未知語の例としては「Siri」「あっちゃん」「先っちょ」「スンゲー」「ドｍ」「花立山」「えらそう」「やべえええええ」などが挙げられる．

表 8.3: 未知語獲得数

未知語候補抽出数	227,367
未知語獲得数	138,546

また，このように語彙の収集にクラウドソーシングを用いることによって時間コストと費用を大きく削減することが出来た．比較対象として，4 万語の語彙を取得し，読みつけを行うまでのコストを比較した．従来の手法として派遣作業者 3 名で単語判定を行い，読みつけを行うケースと，クラウドソーシングを用いて同数の語彙を処理した場合の比較は図 8.8 のようになる．図に示されるように，時間コストの面では 50 日から 5 日と 90%削減が可能となり，コスト比較では 170 万円から 50 万円と 70%削減が可能となった．このように効率の面からもクラウドソーシングの有効性が高いことがわかる．

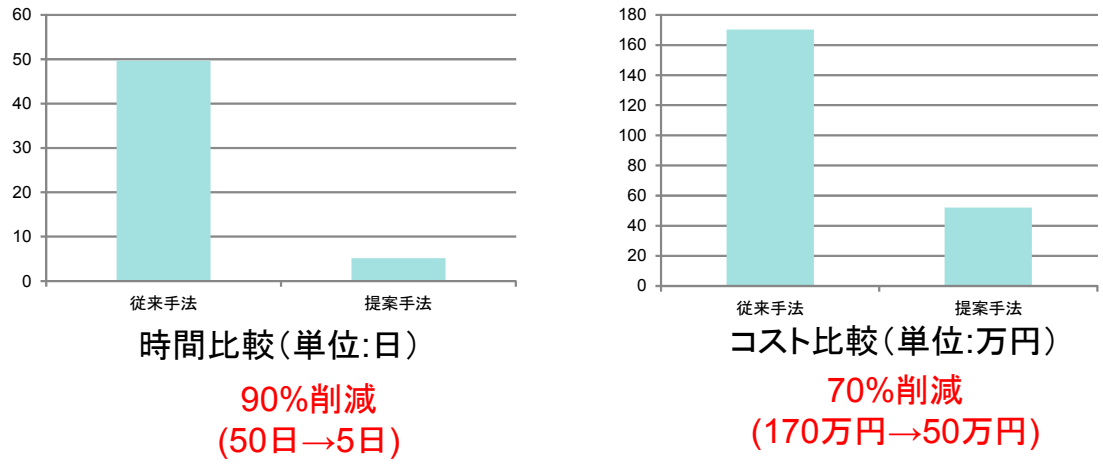

60
50
40
30
20
10
0
従来手法
提案手法
時間比較（単位:日）
90%削減
(50日→5日)
180
160
140
120
100
80
60
40
20
0
従来手法
提案手法
コスト比較（単位:万円）
70%削減
(170万円→50万円)

図 8.8: コスト削減効果

第9章　結論

9.1　まとめ

本研究では研究データを安価，高速かつ高精度に作成するためのマイクロタスク型クラウドソーシングに関する研究である．独自の精度向上手法を用いるためにプライベートなクラウドソーシングシステム（PCSS）を構築し，システム内に様々な精度向上手法を組み込むことで既存のマイクロタスク型クラウドソーシングサービスでは困難なシステム面からの精度向上を実現することが出来た．PCSS で用いた精度向上手法は以下の 2 点に大別できる．

1. ワーカーの行動履歴やタスクの特徴を用いたフィルタリング
2. ベイジアンネットワークを用いて導出した学習手法による段階的学習方法

(1) は以下の 4 つのフィルタリングによって実施されている．

(1)-a　事前フィルタリング

ワーカーを募集する際にベースとなる能力やモチベーション，プロフィールなどでフィルタリングを行う．また，高難易度のタスクの場合はそのタスクの小規模テストを実施してフィルタリングを行う．このフィルタリングにより最低限のワーカー品質を確保することができる．

(1)-b　動的フィルタリング

ワーカーが作業する過程で正解率を常に計算し，一定以下になったワーカーに該当するタスクカテゴリを割り当てないようにする．全ての平均値が一定値を下回るなどスパムワーカーと判断できる場合はシステムへのログインを禁止するなどで排除する．

(1)-c　結果フィルタリング

タスクの処理結果からワーカー毎の得意，不得意な分野を解析し，得意分野に所属するタスクを優先的に割り当て，不得意分野に所属するタスクを割り当てないようにすることで最適なタスクアロケーションを行う．

(1)-d　推測フィルタリング

ワーカーの作業履歴からワーカーの特性を解析し，ワーカー間の類似性を解析して協調フィルタリングを用いることで，まだ未着手のタスクにおける得意，不得意を推測する．

また，(2) は以下の 2 つのステップで実施されている．

(2)-a タスクグループのカテゴリ分類

タスクにおけるタイトルや説明文から TFIDF 値を求め，コサイン類似度を計算することで類似した内容のタスクをタスクカテゴリとして管理する．

(2)-b タスクカテゴリ間の関係性の解析

得られたタスクカテゴリにおけるワーカーの精度を用いてベイジアンネットワークにてタスク間の有向グラフを作成し，影響のあるタスクを学習タスクとして用いる．

このような精度向上手法を用いた PCSS 用いて研究データの作成を行っている．研究データの構築の実例として自然言語処理の研究に用いる未知語の収集を示した．Web からクローラを用いて日本語 125 億文の Web テキストを収集し，形態素解析で得られた未知語候補に対して PCSS を用いることで 14 万語の未知語を抽出することに成功した．また，この未知語抽出の過程で (1) の精度改善のためのフィルタリングを適用することで，平均 32.4 ポイントの精度改善効果を得ることが出来た．また，同様に未知語抽出の過程で得られたワーカーの行動履歴から (2) の段階的学習手法を用いて学習タスクを導出したところ，9 個の対象となる低精度タスクカテゴリに対して 31 個の学習タスクを導出することに成功し，さらに得られた学習タスクを用いて低品質ワーカーへの学習を行うことによって平均 7.8 ポイントの精度向上効果が得られることを確認できた．

9.2 今後の課題

本研究を通して得られた課題としては以下の点がある．

1. ワーカーのモチベーションコントロール

2. セキュリティ向上

(1) のモチベーションコントロールはマイクロタスク型のクラウドソーシングにおけるコスト，速度，精度，全ての点に影響を与える大きな問題である．マイクロタスク型のクラウドソーシングはマッチング型やコンペティション型のクラウドソーシングと比較して一つ一つの作業が小さいため達成感があまり得られない．そのため何らかの目標を与える必要がある．本研究では難易度や高いモチベーションが必要なタスクにおいては報酬の高低でコントロールしているが，報酬によるモチベーションコントロールはマイクロタスク型クラウドソーシングの低コストという利点を損なう可能性が出てきてしまう．また，段階的学習手法においても学習意欲と言う形でモチベーションは重要であり，学習意欲の低下は学習効果の低下につながる．しかし，ワーカーの学習はシステム側に依存する部分が大きく，リクエスタに学習のための報酬を依頼するのは難しいなどの問題もあり，報酬に依存しないモチベーションの向上手法を検討する必要がある．

報酬を用いないワーカーのモチベーションのコントロールとしてゲーミフィケーション的なアプローチが効果がある [Ahn 08]．クラウドソーシングにおけるゲーミフィケーションの適用は 2.3 で述べているようにタスクの内容に依存するものが多く，本研究におけるシステム側からの適用は難易度が高い．システム側におけるゲーミフィケーションの適用としては以下のような方法が考えられる．

- ワーカー間の競争心を刺激する
  - ランキング設定
  - ライバルワーカーの設置
  - 勝敗ルールの設定，報酬への重み付け
- ワーカー同士で協力してタスクの処理や学習をする

  - ワーカー同士でのグループワークの許可
  - アクティブ・ラーニング

- 適切なマイルストンを設置する
  - レベルやアイテムなどのコストを伴わない報酬設定
  - レベリングによる優遇措置

特にアクティブ・ラーニングや協調学習に関しては既存研究 [Ueno 00] でも触れられており，マイクロタスク型クラウドソーシング上でも有効性が予想されるが，これらの適用はワーカーからの問い合わせやクレームが増加することも予測され，適用には慎重な対応が求められる．

(2) のセキュリティ保持に関してはクラウドソーシングが「不特定多数の作業者による処理」であるため難易度が高い．本研究ではプライベートな環境でシステムを構築し，タスクを分散化するなど対応を行っているが，研究データ以外のデータとして個人情報を処理するケースなど更なる高セキュリティを要する条件も発生しつつある．さらに，クラウドソーシングの有用性が高まるに応じて企業における需要も高まりつつあり，同様にセキュリティ保持に関する需要も高まっている．最も単純な対応としてはワーカーをセキュアな環境下に置いて作業をさせる，ワーカー全てと秘密保持契約（Non-disclosure agreement，NDA） を結ぶなどが考えられるが，コストの面から現実的ではない．

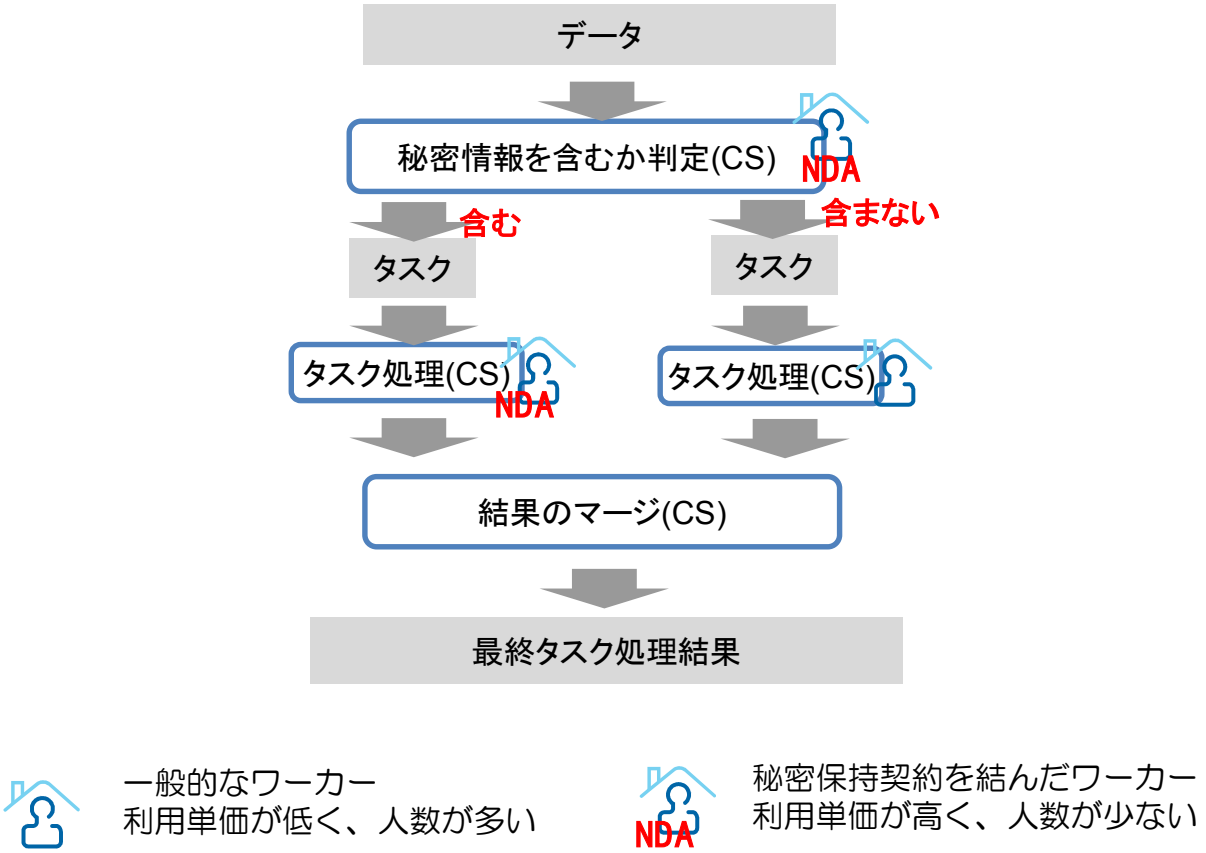


図 9.1: ハイブリッドクラウドソーシング

対策の一つとして図 9.1 のように秘密保持ワーカーと一般ワーカーを組み合わせたハイブリッドな対応が考えられる．秘密保持ワーカーはコストを抑えるために最低限の人数とし，秘密保持が必要なデータのチェックや洗い出しを主に担当する．具体的なタスク処理と比較してチェック関連は低コストで行えることが多いため，秘密保持ワーカーのコストを最低限にして処理することが可能となる．これにより，全ての処理を秘密保持ワーカーが行うよりも高速，低コストで行うことが可能となり，全ての処理を一般ワーカーが行うよりもセキュリティ保持が可能，精度チェックが可能となる．しかし，対象となるデータの全般に秘密保持条件が関連する，リクエスタが秘密保持条件を明確化出来ないなどのケースでは，このハイブリッドクラウドソーシングでは対応できず今後の課題である．

謝辞

本研究にあたり，ご多忙の中適切なご指導をくださった大須賀 昭彦 教授，川村 隆浩 客員准教授に感謝いたします．また，様々な協力をしてくださった大須賀・清研究室＆田原研究室の皆様に感謝の意を表します．そして，投稿した論文等に対して，国内外の多くの査読者から様々なコメントをいただきました．

審査を快く引き受けてくださいました大学院 情報システム学研究科の南 泰浩 教授，植野 真臣 教授，田原 康之 准教授に感謝申し上げます．先生方には，論文のまとめ方や技術の評価方法などに関して多大なご指導をいただきました．また，ご指導くださった栗原 聡 教授に感謝いたします．

本研究は株式会社 東芝において，多くの方々のご指導とご協力を得て行ったものに基づいています．株式会社 東芝に在籍のまま社会人博士課程への就学を許可いただき便宜を図って戴いた，研究開発センター所長 堀 修 氏，研究開発センター知識メディアラボラトリ室長 出羽 達也 氏，研究開発センター知識メディアラボラトリ主任研究員 池田 朋男 氏，に心から感謝申し上げます．また研究活動を進める上でご指導をくださり，様々なご支援とご配慮を頂きました，研究開発センター知識メディアラボラトリ主務 中田 康太 氏，研究開発センター知識メディアラボラトリ主務 宮村 祐一 氏，研究開発センターの先輩，同僚，後輩の方々に心から御礼申し上げます．

参考文献

[Akasaka 09] Akasaka, R., “Foreign Accented Speech Transcription and Accent Recognition Using a Game-based Approach ”, Masters Thesis, Swarthmore Department of Linguistics, (2009).

[Ahn 08] Ahn, L., Dabbish, L., “Designing games with a purpose”, Communications of the ACM, pp. 58-67, (2008).

[Ahn 15] Ahn, J., Sarah W., and Brian S. B., “Open education in the wild: The dynamics of course production in the peer 2 peer university.”, Proceedings of the 18th ACM conference on computer supported cooperative work & social computing, ACM, pp1896-1905, (2015).

[Ali 10] Ian, E., Ali, F., Derek, H., David A, F., “The Benefits and Challenges of Collecting Richer Object Annotations”, In 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops, pp.1-8, (2010).

[Almond 09] Almond, R., et al., “Bayesian networks: A teacher2̆019s view.” International journal of approximate reasoning 50.3, pp.450-460, (2009).

[Ambati 11] Ambati, V. et al., “Towards task recommendation in micro-task marketss”, Human computation, pp.1-4, (2011).

[Ann 10] Ann, I., Alexandre, K., “Using Mechanical Turk to Annotate Lexicons for Less Commonly Used Languages”, Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon’s Mechanical Turk. Association for Computational Linguistics, pp.108-113, (2010).

[芦川 12] 芦川 将之., 西山 修., 下郡 信宏., ”CrowdSourcing を用いた単語への読み付け, アクセント付け手法の提案”, 電子情報通信学会技術研究報告 ,111(447), pp. 11-16, (2012).

[芦川 13] 芦川 将之., 宮村祐一., 有賀康顕., ”PrivateCrowdSourcing を用いた言語, 音声資源の収集 ～システムの構築と言語収集～”, 人工知能学会全国大会,(第 27 回), (2013).

[Ashikawa 14] M.Ashikawa, T.Kawamura and A.Ohsuga. “Speech Synthesis Data Collection for Visually Impaired Person.” Third AAAI Conference on Human Computation and Crowdsourcing, (2014).

[馬場 13] 馬場 雪乃, 鹿島 久嗣, 木下 慶, 山口 豪志, 秋好 陽介, “機械学習による不適切なクラウドソーシングタスクの検出”, 第 5 回データ工学と情報マネジメントに関するフォーラム, (DEIM), (2013).

[Audhkhasi 11] Audhkhasi, K., Georgiou, P., Narayanan, S. “Accurate Transcription of Broadcast News Speech Using Multiple Noisy Transcribers and Unsupervised Reliability Metrics”, 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp.4980-4983, (2011).

[Bachrach 12] Bachrach, Y., et al., “How to grade a test without knowing the answers—A Bayesian graphical model for adaptive crowdsourcing and aptitude testing.” International Conference on Machine Learning, pp.1183-1190, (2012).

[Bart 10] Bart, M., Francesc, B., Jens, G., Joan, C., Marta R, C., Rafael, B., “Opinion Mining of Spanish Customer Comments with Non-Expert Annotations on Mechanical Turk”, Proceedings of the NAACL HLT 2010 workshop on Creating speech and language data with Amazon’s mechanical turk. Association for Computational Linguistics, pp.114-121, (2010).

[Bragg 14] Bragg, J., Weld, DS., “Crowdsourcing multi-label classification for taxonomy creation.” First AAAI conference on human computation and crowdsourcing, (2013).

[Bucholz 11] Bucholz, S., Latorre, J., "Crowdsourcing preference tests and how to detect cheating ", Interspeech, pp.3053-3056, (2011).

[Burnap 13] Burnap, A., et al., "A simulation based estimation of crowd ability and its influence on crowdsourced evaluation of design concepts." ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. American Society of Mechanical Engineers, pp.V03BT03A004-V03BT03A004, (2013).

[Butz 06] Butz, C., Hua, S., Maguire, B., "A Web-based Bayesian Intelligent Tutoring System for Computer Programming" Web Intelligence and Agent Systems, IOS Press, Vol.4, No.1 pp.77-97, (2006).

[Carpenter 11] Carpenter, B., "A Hierarchical Bayesian Model of Crowdsourced Relevance Coding." TREC, (2011).

[Cyrus 10] Cyrus, R., Peter, Y., Micah, H., Julia, H., "Collecting Image Annotations Using Amazon ' s Mechanical Turk", Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk. Association for Computational Linguistics, pp.139-147, (2010).

[Dan 10] Dan, G., Yang, L., "Non-Expert Evaluation of Summarization Systems is Risky", Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk. Association for Computational Linguistics, pp.148-151, (2010).

[Dawid 79] Dawid, A.P., Skene,A.M.,"Maximum Likelihood Estimation of Observer Error-Rates Using the EM Algorithm",Journal of the Royal Statistical Society, pp.20-28, (1979).

[Dillahunt 16] Dillahunt, T. R., et al, "Do Massive Open Online Course Platforms Support Employability?.", Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing, ACM, pp233-244, (2016).

[Dongqing 10] Dongqing, Z., Ben, C., "An Analysis of Assessor Behavior in Crowdsourced Preference Judgments", SIGIR 2010 workshop on crowdsourcing for search evaluation, pp.17-20, (2010)

[Donmez 09] Donmez, P. et al., "Efficiently learning the accuracy of labeling sources for selective sampling.", Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, pp.259-268, (2009).

[Dorn 15] Dorn, B., Larissa B. S., and Adam S., "Piloting TrACE: Exploring spatiotemporal anchored collaboration in asynchronous learning.", Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing, ACM, pp393-403, (2015).

[Evanini 10] Evanini, K., Higgins, D., Zechner, K., "Using Amazon Mechanical Turk for transcription of non-native speech", Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk. Association for Computational Linguistics, pp.53-56, (2010).

[Fernandez 11] Fernandez, A., et al., "A system for relevance analysis of performance indicators in higher education using Bayesian networks." Knowledge and information systems 27.3, pp.327-344, (2011).

[Gelas 11] Gelas, H., Abate, ST, Besacier, L., Pellegrino, F., "Quality assessment of crowdsourcing transcriptions for African languages", INTERSPEECH, pp.3065-3068, (2011).

[Garcia 07] Garcia, P., et al., "Evaluating Bayesian networks ' precision for detecting students ' learning styles." Computers & Education 49.3 pp.794-808, (2007).

[Glassman 16] Glassman, E. L., et al. "Learnersourcing Personalized Hints.", Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing, ACM, pp1626-1636, (2016).

[Goel 16] Goel, A. K., and David A. J., “Design of an Online Course on Knowledge-Based AI.”, Thirtieth AAAI Conference on Artificial Intelligence, (2016).

[Gil 16] Gil, Y., “Teaching Big Data Analytics Skills with Intelligent Workflow Systems.”, Thirtieth AAAI Conference on Artificial Intelligence, (2016).

[Goto 11] Goto, M., Ogata, J., “PodCastle: Recent Advances of a Spoken Document Retrieval Service Improved by Anonymous User Contributions ”, INTERSPEECH, pp.3073-3076, (2011).

[Halpin 12] Halpin, H., Blanco, R.,“Machine-Learning for Spammer Detection in Crowd-Sourcing”, Workshop on Human Computation at AAAI, Technical Report WS-12-08, pp.85-86, (2012)

[Hoogerheide 12] Hoogerheide, L., Block, JH., Thurik, R., “Family background variables as instruments for education in income regressions: A Bayesian analysis.” Economics of Education Review 31.5, pp.515-523, (2012).

[Hutton 12] Hutton, A., Liu, A., Martin, CE., “Crowdsourcing Evaluations of Classifier Interpretability.” AAAI Spring Symposium: Wisdom of the Crowd, (2012).

[Ian 10] Ian, S., Graham, T., George, W., Christoph, B., “Hands by hand: crowd-sourced motion tracking for gesture annotation”, 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops. IEEE, pp.17-24, (2010).

[Jeanne 13] Jeanne, P., Daniela, B., Michael, Tjalve., Jieun, O., “Evaluating Voice Quality and Speech Synthesis Using Crowdsourcing”, International Conference on Text, Speech and Dialogue. Springer Berlin Heidelberg, Springer, pp.233-240, (2013).

[John 12] John, S., Alexey, T., Charlie, C., Sarah, D., “A Crowdsourcing Approach to Labelling a Mood Induced Speech Corpus”, Proc. 4th Int. Workshop Corpora Res. Emotion Sentiment Social Signals, (2012).

[Jurcicek 11] Jurcicek, F., Keizer, S., Gasic, M., Mairesse, F., Thomson, B., Yu, K., Young, S., “Real User evaluation of spoken dialogue systems using Amazon Mechanical Turk”, In Proceedings of INTERSPEECH (Vol. 11), (2011).

[Kamar 12] Kamar, E., Hacker, S., Horvitz, E., “Combining human and machine intelligence in large-scale crowdsourcing.” Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 1. International Foundation for Autonomous Agents and Multiagent Systems, pp.467-474, (2012).

[川合 88] 川合 治男, “マイコンによる CAI の実践と課題: アメリカの実践に学ぶ.”, 教育方法学研究 8, pp165-188, (1988).

[Kazai 11] Kazai, G., Kamps,J., Koolen,M., Milic-Frayling, N., “Crowdsourcing for book search evaluation: impact of hit design on comparative system ranking”, Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval. ACM, pp.205-214, (2011).

[Kilian 12] Kilian, N., Krause, M., Runge, N., Smeddinck, J., “Predicting Crowd-Based Translation Quality with Language-Independent Feature Vectors”, HCOMP, (2012)

[Kittur 08] Kittur, A., Chi, E., Suh, B., “Crowdsourcing user studies with mechanical turk”, In Proceedings of the SIGCHI conference on human factors in computing systems, pp.453-456, (2008).

[Kunath 10] Kunath, S.A., and Weinberger, S.H., “The wisdom of the crowd’s ear: speech accent rating and annotation with Amazon Mechanical Turk”, Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon’s Mechanical Turk. Association for Computational Linguistics, pp.168-171, (2010).

[小山 13] 小山 聡, 馬場 雪乃, 櫻井 祐子, 鹿島 久嗣, “クラウドソーシングにおけるワーカーの確信度を用いた高精度なラベル統合”, 人工知能学会全国大会, (第 27 回), (2013).

[Lee 11] Lee, C., Glass, J., “A Transcription Task for Crowdsourcing with Automatic Quality Control”, Interspeech,pp. 3041-3044, (2011).

[Li 15] Li, X., et al, “Massive open online proctor: Protecting the credibility of MOOCs certificates.”, Proceedings of the 18th ACM Conference on Computer Supported Co-operative Work & Social Computing, ACM, pp1129-1137, (2015).

[Luke 09] Luke, B., Douglas, T., Damien, O., Gert, L “Designing A Social Game to Tag Music”, Proceedings of the acm sigkdd workshop on human computation, pp.7-10, (2009)

[Lin 12] Lin, CH., Weld, D., “Crowdsourcing control: Moving beyond multiple choice.” Uncertainty in Artificial Intelligence, pp.491-500, (2012).

[Mao 12] Mao, A., Procaccia, A., Chen, Y., “Social Choice for Human Computation”, HCOMP-12: Proc. 4th Human Computation Workshop, (2012).

[Matthew 10] Matthew, M., Satanjeev,B., Alexander,I., “Using the Amazon Mechanical Turk to Transcribe and Annotate Meeting Speech for Extractive Summarization”, Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon’s Mechanical Turk. Association for Computational Linguistics, pp.99-107, (2010).

[May 06] May, H., “A multilevel Bayesian item response theory method for scaling socioeconomic status in international studies of education.” Journal of Educational and Behavioral Statistics 31.1, pp.63-79, (2006).

[松原 13] 松原繁夫, 水島拓也, “クラウドソーシングにおける複数タスク割当て”, 人工知能学会全国大会, (第 27 回), (2013).

[宮川 04] 宮川雅巳., “統計的因果推論”, 朝倉書店, (2004).

[Mohammad 10] Mohammad, S., Martha, L., ”Crowdsourcing for Affective Annotation of Video: Development of a Viewer-reported Boredom Corpus”, In Proceedings of

the ACM SIGIR 2010 workshop on crowdsourcing for search evaluation (CSE 2010), pp.4-8, (2010).

[本村 11] 本村陽一, “ベイジアンネットワーク.” 電子情報通信学会誌 83.8, pp. 645-646, (2000).

[森 11] 森 信介 , 中田 陽介 , Neubig Graham , 河原 達也, “点予測による形態素解析” , 自然言語処理, Vol.18, no. 4, pp. 367-381, (2011).

[西 13] 西 智樹, 小出 智士, 大野 宏司, 長屋 隆之, “ソーシャルネットワークを用いたクラウドソーシングの品質向上”, 人工知能学会全国大会 (第 27 回), (2013).

[Novotney 10] Novotney, S., Callison-Burch, C., “Cheap, Fast and Good Enough : Automatic Speech Recognition with Non-Expert Transcription ”, Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics. Association for Computational Linguistics, pp207-215, (2010).

[Nushi 15] Nushi, B., et al., “Crowd Access Path Optimization: Diversity Matters.” Third AAAI Conference on Human Computation and Crowdsourcing, (2015).

[Okamoto 08] Okamoto, T., Kayama M., “人工知能と教育工学”, オーム社, (2008).

[Pardos 10] Pardos, ZA., et al., “Using fine-grained skill models to fit student performance with Bayesian networks.” Handbook of educational data mining pp.417-425, (2010).

[Raykar 12] Raykar, V., Yu, S., “Eliminating spammers and ranking annotators for crowdsourced labeling tasks.”, Journal of Machine Learning Research, 13(Feb), pp491-518, (2012).

[Raykar 14] Raykar, VC., Agrawal, P., “Sequential crowdsourced labeling as an epsilon-greedy exploration in a Markov Decision Process.” AISTATS, pp.832-840,(2014).

[Reotutar 16] Reotutar, C., et al. “An Online Logic Programming Development Environment.”, Thirtieth AAAI Conference on Artificial Intelligence, (2016).

[Robert 10] Robert, M., “Crowdsourced translation for emergency response in Haiti: the global collaboration of local knowledge”, AMTA Workshop on Collaborative Crowdsourcing for Translation, pp.1-4, (2010).

[櫻井 12] 櫻井 祐子, 沖本 天太, 岡雅 晃, 兵藤 明彦, 篠田 正人, 横尾真, “クラウドソーシングにおける品質コントロールの一考察”, Joint Agent Workshop and Symposiums, (2012).

[Shaw 11] Shaw, AD., Horton, JJ., Chen, DL., “Designing incentives for inexpert human raters.” Proceedings of the ACM 2011 conference on Computer supported cooperative work. ACM, pp.275-284, (2011).

[Simpson 15] Simpson, E., Roberts, S., “Bayesian methods for intelligent task assignment in crowdsourcing systems.” Decision Making: Uncertainty, Imperfection, Deliberation and Scalability. Springer International Publishing, pp.1-32, (2015).

[Singh 16] Singh, S., and Sebastian R., “Creating Interactive and Visual Educational Resources for AI.”, Thirtieth AAAI Conference on Artificial Intelligence, (2016).

[Snow 08] Snow, R., O’Connor, B., Jurafsky,D., Ng, A.Y., “Cheap and Fast But is it Good? Evaluating Non-Expert Annotations for Natural Language Tasks”, Proceedings of the conference on empirical methods in natural language processing. Association for Computational Linguistics, pp.254-263, (2008).

[Sintov 16] Sintov, N., et al., “From the Lab to the Classroom and Beyond: Extending a Game-Based Research Platform for Teaching AI to Diverse Audiences.”, Symposium on Educational Advances in Artificial Intelligence (EAAI), (2016).

[Sun 12] Sun, Y., Dance, C., "When majority voting fails: Comparing quality assurance methods for noisy human computation environment." Computing Research Repository, vol. 1204.3516, (2012).

[Tae 10] Tae, Y., Philip, R., Noah A, Smith., "Shedding (a Thousand Points of) Light on Biased Language", roceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk. Association for Computational Linguistics, pp.152-158, (2010).

[Tang 11] Tang, W., Lease, M., "Semi-supervised consensus labeling for crowdsourcing." SIGIR 2011 workshop on crowdsourcing for information retrieval (CIR),pp.1-6, (2011).

[Ueno 00] Ueno, M., "Intelligent tutoring system based on belief networks", International Workshop on Advanced Learning Technologies, pp141-142, (2000).

[Vaughan 13] Vaughan, J. W., "Adaptive Task Assignment for Crowdsourced Classification, In 30th Intl. Conf. on Machine Learning (ICML), (2013).

[Venanzi 15] Venanzi, M., et al., "The ActiveCrowdToolkit: An Open-Source Tool for Benchmarking Active Learning Algorithms for Crowdsourcing Research." Third AAAI Conference on Human Computation and Crowdsourcing, (2015).

[Wanf 11] Wang, J., Yu, B., "Labeling Images with Queries A Recall-based Image Retrieval Game", in Proceedings of the SIGIR 2011 Workshop on Crowdsourcing for Information Retrieval, (2011).

[Wais 11] Wais, P., et al., "Towards large-scale processing of simple tasks with mechanical turk.", Third AAAI Conference on Human Computation and Crowdsourcing,(2011).

[Wauthier 11] Wauthier, FL., Jordan, MI., "Bayesian bias mitigation for crowdsourcing." Advances in Neural Information Processing Systems, pp.1800-1808, (2011).

[Welinder 10] Welinder, P., Branson, S., Belongie, S., Perona, P., “The Multidimensional Wisdom of Crowds”, Advances in neural information processing systems, pp.2424-2432, (2010).

[Whitehill 09] Whitehill, J., Ruvolo, P., Wu, T., Bergsma, J., Movelan, J., “Whose Vote Should Count More: Optimal Integration of Labels from Labelers of Unknown Expertise”, Advances in neural information processing systems, (2009).

[Wolters 10] Wolters, M., Isaac, K., Renals, S. “Evaluating Speech Synthesis Intelligibility using Amazon Mechanical Turk, In Proc. 7th Speech Synthesis Workshop (SSW7), (2010).

[Xenos 04] Xenos, M., “Prediction and assessment of student behaviour in open and distance education in computers using Bayesian networks.” Computers & Education 43.4, pp.345-359, (2004).

[Xie 15] Xie, H., Lui, JCS., Towsley, D., “Incentive and reputation mechanisms for online crowdsourcing systems.” Quality of Service (IWQoS), 2015 IEEE 23rd International Symposium on. IEEE, pp207-212, (2015).

[Yang 10] Yang, Z., Li, B., Zhu,Y., King, I., Levow, G. and Meng , H., “Collection of User Judgments on Spoken Dialog System with Crowdsourcing”, Spoken Language Technology Workshop (SLT), pp.277-282, (2010).

[Yano 13] ＢＰＯ市場クラウドソーシング市場に関する調査結果 2013, 矢野経済研究所, (2013).

[Yoon 16] Yoon, D., et al., “RichReview++: Deployment of a Collaborative Multi-modal Annotation System for Instructor Feedback and Peer Discussion.”, Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing, ACM, pp195-205, (2016).

[Yuen 12] Yuen, M. C., et al., "TaskRec: probabilistic matrix factorization in task recommendation in crowdsourcing systems", International Conference on Neural Information Processing. Springer Berlin Heidelberg, pp516-525, (2012).

[Zagalsky 15] Zagalsky, A., et al., "The emergence of github as a collaborative platform for education.", Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing, ACM, pp1906-1917, (2015).

[Zheng 16] Zheng, S., et al., "Ask the Instructors: Motivations and Challenges of Teaching Massive Open Online Courses.", Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing, ACM, pp206-221, (2016).

研究業績

学術雑誌

1. 芦川 将之, 川村 隆浩, 大須賀 昭彦： マイクロタスク型クラウドソーシングプラットフォーム環境における精度向上手法の導入と評価, 人工知能学会論文誌, 29(6), pp.503-515, 2014 年 1 月.

2. **Masayuki Ashikawa**, Takahiro Kawamura, Akihiko Ohsuga : Quality Improvement by Worker Filtering and Development in Crowdsourcing, Web Intelligence, Journal, IOS press, vol. 14, no. 3, pp. 229-244, 2016 年 8 月.

3. 芦川 将之, 川村 隆浩, 大須賀 昭彦： クラウドソーシングワーカーの段階的育成方法の提案, 人工知能学会論文誌, 32(3), 2017 年 5 月 (採録決定済).

国際会議

4. **Masayuki Ashikawa**, Takahiro Kawamura, Akihiko Ohsuga: Deployment of Private Crowdsourcing System with Quality Control Methods, 2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT), pp.9-16. Vol. 1. IEEE, December, 2015.

5. **Masayuki Ashikawa**, Takahiro Kawamura, Akihiko Ohsuga: Speech Synthesis Data Collection for Visually Impaired Person, Second AAAI Conference on Human Computation and Crowdsourcing (HCOMP 2014), November 2014. （ワークショップ）

6. **Masayuki Ashikawa**, Takahiro Kawamura, Akihiko Ohsuga: Proposal of Grade Training Method in Private Crowdsourcing System, Third AAAI Conference on Hu-

man Computation and Crowdsourcing (HCOMP 2015), pp.2-3, AAAI Press, November 2015. （ポスター）

国内大会・研究会

7. 酒井 敏彦, 芦川 将之，廣川佐千男: Crowdsourcing System を用いた略語の推定手法の提案，電子情報通信学会技術研究報告．NLC, 言語理解とコミュニケーション, 111(364), pp.13-17, (2011)

8. 芦川 将之，西山 修，下郡 信宏: CrowdSourcing を用いた単語への読み付け, アクセント付け手法の提案，電子情報通信学会技術研究報告．AI, 人工知能と知識処理, 111(447), pp.11-16, (2012)

9. 芦川 将之，有賀 康顕，宮村 祐一: PrivateCrowd-Sourcing を用いた言語, 音声資源の収集 システムの構築と言語収集，人工知能学会全国大会, (第 27 回), (2013)

10. 芦川 将之，川村 隆浩，大須賀 昭彦: プライベートクラウドソーシングにおける精度向上手法，人工知能学会全国大会論文集 28, pp.1-4, (2014)

11. 芦川 将之，川村 隆浩，大須賀 昭彦: クラウドソーシングワーカーの段階的育成方法の提案，人工知能学会全国大会論文集 29, pp.1-4, (2015)

解説記事

- 芦川 将之，池田　朋男: クラウドソーシングを用いたアノテーション (＜特集＞ ヒューマンコンピュテーションとクラウドソーシング)，人工知能: 人工知能学会誌, 29(1), pp.54-59, (2014)

新聞記事

- コアテクノロジー・人工知能＆ビッグデータ活用／東芝，データ識別性能を向上: 日刊工業新聞, 2015 年 10 月 5 日, (2014)

登録特許

- 芦川　将之, 宮村　祐一, 有賀　康顕: 評価値算出装置，評価値算出方法およびプログラム, 特許第 5813845 号, 登録 2015 年 10 月.

関連論文の印刷公表の方法及び時期

学術雑誌

1. 全著者名： 芦川 将之, 川村 隆浩, 大須賀 昭彦
   論文題目： マイクロタスク型クラウドソーシングプラットフォーム環境における精度向上手法の導入と評価
   印刷公表の方法及び時期：人工知能学会論文誌, 29(6), pp.503-515, 2014 年 1 月.
   (第 3,4,5,8 章と関連)

2. 全著者名： **Masayuki Ashikawa**, Takahiro Kawamura, Akihiko Ohsuga
   論文題目： Quality Improvement by Worker Filtering and Development in Crowdsourcing
   印刷公表の方法及び時期： Web Intelligence, vol. 14, no. 3, pp. 229-244, 2016 年 8 月
   (第 3,4,5,6,7,8 章と関連)

3. 全著者名： 芦川 将之, 川村 隆浩, 大須賀 昭彦
   論文題目： クラウドソーシングワーカーの段階的育成方法の提案
   印刷公表の方法及び時期：人工知能学会論文誌, 32(3), 2017 年 5 月 (採録決定済).
   (第 6,7 章と関連)

国際会議

4. 全著者名：**Masayuki Ashikawa**, Takahiro Kawamura, Akihiko Ohsuga
   論文題目：Deployment of Private Crowdsourcing System with Quality Control Methods
   印刷公表の方法及び時期： 2015 IEEE/WIC/ACM International Conference on Web

Intelligence and Intelligent Agent Technology (WI-IAT), pp.9-16. Vol. 1. IEEE, December, 2015. (第 3,4,5,8 章と関連)

5. 全著者名：**Masayuki Ashikawa**, Takahiro Kawamura, Akihiko Ohsuga
論文題目：Speech Synthesis Data Collection for Visually Impaired Person.
印刷公表の方法及び時期：Second AAAI Conference on Human Computation and Crowdsourcing (HCOMP 2014), Citizen + X: Workshop on Volunteer-based Crowdsourcing in Science, Public Health and Government (ワークショップ発表), November 2014.
(第 3,4,5,8 章と関連)

6. 全著者名 **Masayuki Ashikawa**, Takahiro Kawamura, Akihiko Ohsuga
論文題目：Proposal of Grade Training Method in Private Crowdsourcing System.
印刷公表の方法及び時期：Third AAAI Conference on Human Computation and Crowdsourcing (HCOMP 2015), AAAI Press, pp.2-3, (ポスター発表), November 2015.
(第 6,7 章と関連)

本研究との関連の詳細

章	関連論文番号	関連する内容
3 章	1,2,4,5	PCSS の構築
4 章	1,2,4,5	ワーカーフィルタリング手法による精度向上手法の提案
5 章	1,2,4,5	ワーカーフィルタリング手法の評価及び考察
6 章	2,3,6	段階的学習手法による精度向上手法の提案
7 章	2,3,6	段階的学習手法の評価及び考察
8 章	1,2,4,5	PCSS を用いた語彙収集

著者略歴

芦川 将之（あしかわ　まさゆき）

1976 年 7 月 31 日	東京都豊島区に生まれる
1995 年 3 月	私立早稲田高等学校 卒業
1995 年 4 月	私立早稲田大学 理工学部 情報学科 入学
1999 年 3 月	私立早稲田大学 理工学部 情報学科 卒業
1999 年 4 月	私立早稲田大学 大学院 理工学研究科 情報工学専攻 博士前期課程 入学
2001 年 3 月	私立早稲田大学 大学院 理工学研究科 情報工学専攻 博士前期課程 卒業
2001 年 4 月	株式会社 東芝 入社
2014 年 10 月	国立大学法人 電気通信大学 大学院 情報システム学研究科 社会知能情報学専攻 博士後期課程 入学
2017 年 3 月	国立大学法人 電気通信大学 大学院 情報システム学研究科 社会知能情報学専攻 博士後期課程 修了予定