マイクロタスクでの自己補正におけるワーカの回答パターン分析

小林　正樹†　　松原　正樹††　　森田ひろみ††　　清水　伸幸†††　　森嶋　厚行††

† 筑波大学 図書館情報メディア研究科　〒 305-8550 茨城県つくば市春日 1-2
†† 筑波大学 図書館情報メディア系　〒 305-8550 茨城県つくば市春日 1-2
††† ヤフー株式会社　〒 102-8282 東京都千代田区紀尾井町 1-3 東京ガーデンテラス紀尾井町 紀尾井タワー
E-mail: †makky@klis.tsukuba.ac.jp, ††{masaki,morita,mori}@slis.tsukuba.ac.jp, †††nobushim@yahoo-corp.jp

あらまし　マイクロタスクにおける自己補正は，２段階での回答と参考情報の提示を組み合わせたタスク設計による，タスク結果の品質改善のための手法の１つである．自己補正では，タスクの第１段階から第２段階にかけてどのように回答を変更したか，というワーカの行動を捉えることが出来る．既存のワーカの品質推定手法では，回答既知タスクやタスク処理時間などの特徴に注目するのが一般的であったが，自己補正におけるワーカの行動は新たなワーカの特徴として活用できる可能性がある．自己補正におけるワーカの行動から，信頼できるワーカの予測が出来れば，そのようなワーカへの積極的な動機付け等への応用が期待できる．そこで本研究では，自己補正タスクに取り組むワーカの回答行動のパターンに着目することで，信頼性の高いワーカの予測を試みた．実験結果は，複数の自己補正タスクにおける回答の変更頻度と，ワーカから得られるタスク結果の品質の関連性を示唆するものである．具体的には，自己補正タスクを繰り返すことで正答率が改善したワーカには，回答の変更頻度が高すぎず，かつ低すぎない傾向が見られた．

キーワード　クラウドソーシング

1　はじめに

クラウドソーシングは，人間の作業と計算機ネットワークによる情報処理を組み合わせることで様々な問題に取り組む手法である．作業の依頼者であるリクエスタが，不特定多数の作業者であるワーカに対して作業であるタスクを依頼するのが基本的な枠組みである．本稿では，画像や映像，音声へのタグ付けや分類，文章校正などの作業を扱うマイクロタスク型クラウドソーシングに注目する．

クラウドソーシングにおいて，成果物の品質を保証することが重要な研究課題の１つである．成果物の品質が低くなる要因としては，人間による作業が伴うことから成果物の一部に誤答が含まれる可能性や，ランダムな回答により単に報酬を受け取ることを目的とするスパムワーカの存在が挙げられる．これまでに多くの研究がこの問題に取り組んでおり，マイクロタスク型クラウドソーシングの大部分を占めると考えられる分類タスクやタグ付けタスクでは，主に次の３つのアプローチが用いられる．１つ目は優れたワーカを発見し，彼らに対してタスクを割り当てる方法である．例えば，Amazon Mechanical Turk ではリクエスタからの評価が高いワーカに対して作業を割り当てる MTurk Master Worker と呼ばれる仕組み [1]を利用することが出来る．２つ目は，同じタスクを複数のワーカに割り当て，複数のタスク結果を集約することである．最も単純な方法としては多数決が挙げられるが，ワーカやタスクの性質を考慮した様々な手法が提案されている．３つ目は，個々のワーカからより良い結果を引き出す方法である．Shah らは，ワーカがタスクに回答したした後に，同様のタスクに回答した別のワーカの回答を提示し，回答を訂正する機会を与える自己補正と呼ばれる手法を提案した [1]．自己補正はリクエスタがタスク画面を編集できる機能を持つ一般的なクラウドソーシングプラットフォームにおいて，タスクに適用が可能である．

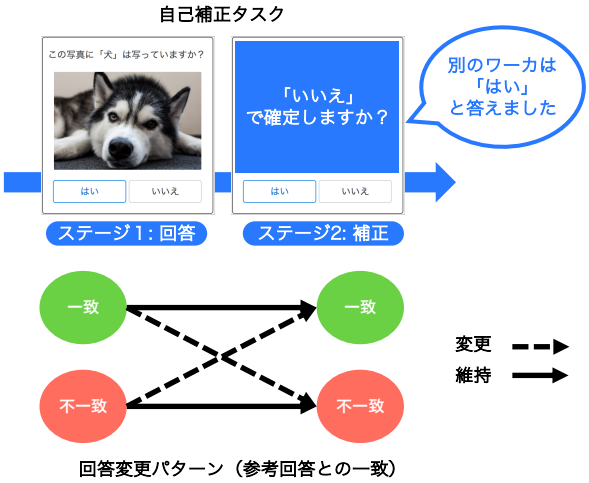

図１　自己補正タスクにおける回答パターン

Shah らは，ワーカのステージ１での成績が低い場合に，特に自己補正が有効であると主張した．ここで重要なのは，ワーカがステージ１での誤りに気づくことが出来た場合に，ステージ２においてステージ１の誤答を訂正出来ることである．つまり，タスクに自己補正を導入することで，ワーカに彼ら自身の誤りに気づかせる機会を与えることが出来るのである．

自己補正の有効性は，Shah らの提案論文におけるシミュレー

1：https://www.mturk.com/worker/help#what_is_master_worker

ションおよび小林ら [2] の実験により示されている．小林らは，Shah らの主張であるマイクロタスクに自己補正を適用することによるタスク結果の改善に加えて，ワーカが自己補正タスクを繰り返すことで学習効果が見られたと報告した．

自己補正の第一段階で誤答をしたワーカが，第 2 段階で提示された参考回答が正しいと判断した場合，ワーカは第 2 段階での回答を変更すると考えられる．このような参考回答を活用して適切な判断を行うことが出来たワーカは，自己補正を繰り返すことにより正答率の改善が期待できる．一方で，単に提示された他者回答を追従したり，全く回答を変更しないワーカも存在する可能性が考えられ，そのようなワーカには学習が見られないだろう．このような自己補正におけるワーカの行動に基づいて，ワーカの信頼性を予測できる可能性がある．しかし，自己補正におけるワーカの振る舞いについて，既存研究では示されていない．

そこで本研究では，自己補正タスクに取り組むワーカの行動データからのワーカの信頼性の予測に取り組む（図 1）．実験結果は，複数の自己補正タスクにおける回答の変更頻度と，ワーカから得られるタスク結果の品質の関連性を示唆するものである．

本研究における主な実験結果は次のとおりである．

（1） 自己補正タスクを繰り返すことで回答品質が改善するワーカには，回答の変更頻度が高すぎず，かつ低すぎない傾向が見られた．

（2） このような回答変更率と正答率の関係は参考回答として正答を提示する条件でのみ見られ，ランダムな回答を提示する条件では見られなかった．

（3） 回答の変更頻度が高いワーカは，回答の変更頻度が低いワーカと比べて，第一段階で選ぶ回答の品質が低い傾向が見られた．そして，回答の変更頻度が高くなるにつれてワーカの第 2 段階での正答率は提示された参考回答の正答率に近づく．この傾向は自己補正の参考回答の条件にかかわらず見られた．

2 関 連 研 究

ワーカへのフィードバックに着目したさまざまな研究があり，フィードバックによってワーカから得られるタスク結果の品質が向上することが知られている．Revolt [3] や Microtalk [4] ではあるワーカの回答を別のワーカが評価し，その評価を確認した上で回答を変更する機会を与える仕組みが用いられている．Shepherd はワーカの自己評価と様々な形態の外部評価を組み合わせるクラウドソーシングのためのフィードバックシステムである [5]．Shah らの自己補正では同じタスクに回答した他者の回答を提示するという単純なフィードバックを用いる．しかし，このフィードバックがどのように機能するかは明らかでない．

ワーカによるワーカ自らの評価には偏りがあると知られている．Gadiraju らはクラウドワーカが彼らの実際の能力についての認識に欠けていることが多いことを示した [6]．このようなバイアスを自己補正の枠組みに取り入れることは，自己補正における興味深い課題の 1 つである．

ワーカから得られるタスク結果を改善するために，ワーカの回答精度の改善に注目する場合，ワーカに対して本番のタスクを割り当てる前に訓練タスクを割り当てる手法が広く用いられている．ワーカに訓練タスクを割り当てた後に本番タスクを割り当てることで，本番タスクの品質が改善されることが知られている [7]．このような手法では，リクエスタやクラウドソーシングプラットフォーム運営者が訓練のためのタスクを用意する必要がある．また，ワーカに対して正誤のフィードバックを与える場合にはタスクと対応する正解を事前に得ておく必要がある．鈴木らは，ワーカが作業に必要なスキルを獲得することを支援するために，ワーカに対してインターンとメンターという関係を設けるマイクロインターンシップの仕組みを提案した [8]．

マイクロタスクにおける知覚学習には，ワーカへのフィードバックが重要であると考えられる．Abad らは，誤った回答をしたワーカに対してルールに基づいてフィードバックを与えることが，ワーカの訓練に効果的であることを示した [9]．本研究における関心は，この様なワーカの知覚学習が，自己補正で提示するような単純なフィードバックでも生じるかである．本研究では，ワーカが自己補正を適用したタスクをこなす過程で，ワーカの知覚学習が観察され，ワーカから得られる回答の品質が改善されるかどうかを検証する．知覚学習が生じるための重要な要素として，ワーカが同じ作業を繰り返して行うことが挙げられる．Law らは，ワーカが同じタスクを長時間こなすことを促すためのインセンティブ設計について議論した [10]．自己補正の枠組みにこのような仕組みを導入することは興味深い課題の 1 つである．

自己補正タスクを作成するためには，提示するための参考回答を事前に容易する必要があるが，自己補正の第 2 段階において信頼性の高い回答を提示するための方法は，自明でない．信頼性の高いワーカの回答を提示するために重要となるのが，ワーカの品質を評価する仕組みである．ワーカの品質を評価する仕組みやアルゴリズムについては様々な研究がなされている [11] [12] [13]．最も単純な方法は，ワーカに割り当てるタスクの中に，ワーカの能力を測定するための特別なタスクを追加することである．クラウドソーシングの文脈ではゴールドスタンダードクエスチョンと呼ばれている．より正確にワーカの品質を推定するために，ワーカが作業を開始した直後の数タスクによりワーカの品質推定を行うのではなく，作業の中盤や後半においても継続的にゴールドスタンダードクエスチョンを割り当てることが効果的であると示されている [14]．さらに，ワーカの評価のためにゴールドスタンダードクエスチョンを使用せず，複数のワーカの回答の照合結果からワーカの品質を推定する手法も提案されている [15] [16]．クラウドソーシングでは正解が未知の課題を扱うことが多いことから，これらは有効な手段であると考えられる．

3 実　　験

3.1 実験環境

実験は Yahoo クラウドソーシング [2] と Crowd4U[3] を組み合わせて行った．ワーカの公募と報酬の支払いは Yahoo! クラウドソーシングを通じて行い，実際の作業ページは Crowd4U を用いて作成した．

3.2 実験参加者

Yahoo! クラウドソーシング上で報酬ありの作業として掲載することで参加者を公募した．作業に関する説明は日本語で記述されているため，実験参加者の多くは日本人であるか，日本語が理解できるようなワーカであると想定される．実験に最後まで参加した参加者には，回答の品質に関わらず 100 円相当の報酬を支払った．

3.3 実験で扱う課題

この実験では，絵画を提示し，その絵画が選択肢の項目のどの画家の作品であるかを推定する課題を作成し，用いた．絵画の画像データについては wikiart.org [4] にて収集した．

3.4 タ　ス　ク

実験では，提示された画像が与えられた 4 種類の選択肢のどの項目に該当するかを判断する画像分類課題を扱った．2

ワーカは，提示された画像が選択肢のどの項目に該当するかを推測し，その項目を選ぶ．実験では，自己補正を適用した自己補正タスクと，ワーカの評価のためのテストタスクを組み合わせて用いる．以下では，それぞれのタスクについて詳述する．

a） テストタスク

テストタスクでは，ワーカは与えられた画像に対して単に分類作業を行う．選択肢の各項目の画像またはテキストをクリックすることで，回答とする項目を選ぶことが出来る．選択が済んだ後に，回答ボタンを押すことで，次のタスクへ遷移，または一連の作業が完了する．

b） 自己補正タスク

自己補正タスクにおいても，テストタスクと同様の画像分類課題を行う．自己補正タスクでは，選択肢を選ぶ機会が 2 回与えられる点が異なる．以降では各回答の機会についてステージ 1，ステージ 2 と呼ぶ．ステージ 1 において，選択肢からいずれかの項目を選択し，回答ボタンを押すことで，ステージ 2 の画面が表示される．ステージ 2 では，ステージ 1 でのワーカ自身の回答がラジオボタンに維持されているのに加え，参考回答である項目が赤枠でハイライトされる．ステージ 2 において，ワーカは自身の回答と参考回答を見た上で，最終的な回答を判断することが出来る．

2 : https://crowdsourcing.yahoo.co.jp

3 : https://crowd4u.org

4 : https://www.wikiart.org/

表 1　自己補正タスクにおける回答の変更パターン

		stage1-stage2	
		変更なし	変更あり
参考回答と	一致	a-a	d-a
	不一致	d1-d1	d1-d2,a-d

表 2　実験の構成

	フェーズ	タスクの種類	タスク数
1	Pre テスト	テスト	12
2	学習 1	自己補正	52
3	Mid テスト	テスト	12
4	学習 2	自己補正	52
5	Post テスト	テスト	12

3.5 回答パターン

表 1 に，自己補正タスクにおける回答の変更パターンの種類を示す． 表中の a は参考回答との一致を，d は参考回答との不一致を表している．自己補正タスクにおける回答変更率を式 1 に示す．

$$\text{回答変更率} = \frac{\text{回答変更あり}}{\text{回答変更なし} + \text{回答変更あり}} \tag{1}$$

3.6 比較する条件

実験では，常にランダムな回答を見せる random 条件と，常に正答を提示する correct 条件について，タスク結果の正答率を比較した．実験に参加するワーカはどちらの条件のグループに割り当てられるかについて告知しない．

3.7 実験デザイン

実験でワーカが取り組むタスクの構成を表 2 に示す．ワーカは一連の実験の通して 2 種類で構成される 5 つのフェーズのタスクに順番に回答する． 2 種類のフェーズの 1 つ目は，テストフェーズである．このフェーズではテストタスクが提示される．2 つ目は，学習フェーズである．このフェーズでは自己補正タスクが提示される． 2 つのテスト時期の間に学習フェーズを割り当てることで，学習効果を測定する．フェーズの構成は全てのワーカに対して共通であるが，出題するタスクや出題の順番はワーカごとにランダムに決定した．

3.8 ワーカのフィルタリング

実験では，ワーカをフィルタリングするためのタスクを導入し，それらのタスクに正答できたかどうかに基づいて分析対象とするワーカをフィルタリングする．フィルタリングのためのタスクは各学習フェーズに 2 タスクずつ追加した．タスクでは，質問として選択肢として提示されている画像のうちの 1 つが提示される．ワーカに対してはこれらのタスクが含まれていることは告知せず，フィルタリングにより分析の対象から除外される場合にも報酬を支払った．

4 結　　果

実験参加者 191 名からのデータが得られた．表 3 に，参考

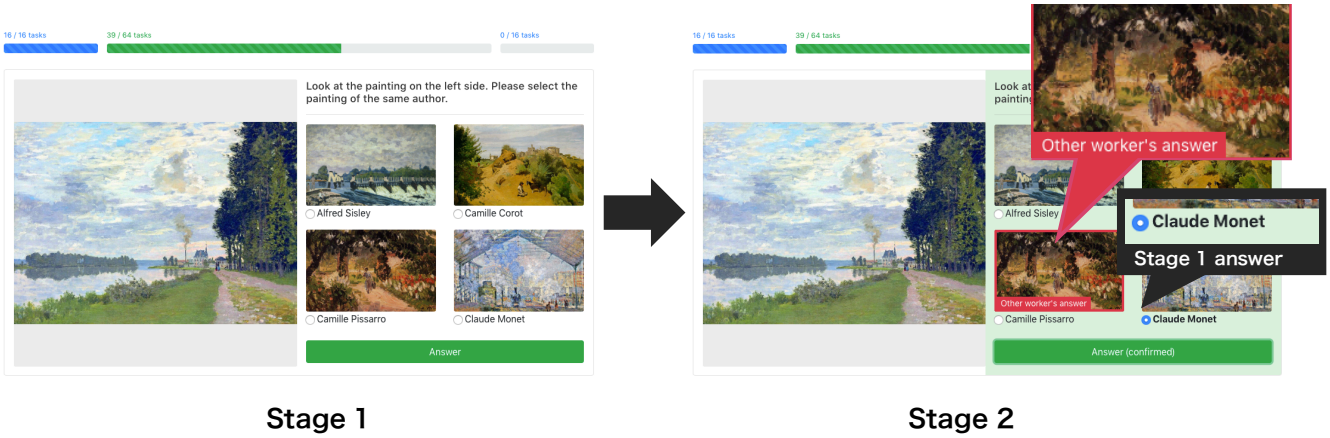

図 2　実験で用いる自己補正タスクの一例

表 3　フィルタリングタスクの正答数と Pre テストの正答率の関係

	参考回答の条件			
	Correct		Random	
タスク数	ワーカ数	Pre テスト	ワーカ数	Pre テスト
0	1	0.083	0	0
1	3	0.25	6	0.25
2	7	0.381	7	0.345
3	17	0.387	18	0.347
4	58	0.353	74	0.365

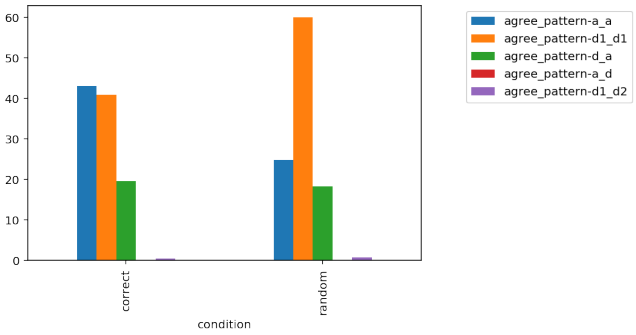

図 3　参考回答の条件ごとの，自己補正タスクの各回答変更パターン

回答の各条件における，フィルタリングタスクの正解数ごとのワーカ数および Pre テストの平均正答率を示す．それぞれの条件において，すべてのフィルタリングタスクに正答したワーカが最も多く，フィルタリングタスクの正答率が低くなるにつれて該当するワーカの人数が減少する傾向が見られた．全体的な傾向として，フィルタリングタスクの正答数が 2 を超えるようなワーカの Pre テストの平均正答率は同様であった．このことから，実験結果の分析では，フィルタリングタスクの正答数が 2 を超えたワーカを分析の対象とした．

4.1　自己補正タスクの回答変更パターン

図 3 に，ステージ 2 での正誤における自己補正の各パターンの平均値を示す．グラフは参考回答の条件毎に別れており，棒の種類が各回答パターンを，横軸がその頻度の平均を表している．

4.2　回答変更率と自己補正の各ステージの正答率

図 4 に回答変更率と自己補正のステージ 1 の正答率の関係を示す．同様に図 5 に，回答変更率とステージ 2 の正答率の関係を示す．これらのグラフの横軸は回答変更率を，縦軸はタスクの正答率を表している．

回答変更率におけるステージ 1 の正答率についての統計的検定をするために，回答変更率を 0.2 ごとに分け，5 つの区間についてステージ 1 の正答率の平均値を比較した．この回答変更率の区間ごとのステージ 1 の正答率を図 6 に示す．Correct 条件における回答変更率の区間でのステージ 1 の正答率に差があるかを検証するために，独立変数を回答率の各区間，従属変数をステージ 1 の正答率とする一要因分散分析を行った．その結果，主効果が認められた ($F(3, 78) = 6.457$, $p < .001$)．Bonferroni 法による多重比較では，0.2-0.4 と 0.6-0.8 の間 ($p < .001$) に有意差が認められた．次に Random 条件における回答変更率の区間でのステージ 1 の正答率に差があるかを検証するために，独立変数を回答率の各区間，従属変数をステージ 1 の正答率とする一要因分散分析を行った．その結果，主効果は認められなかった ($F(4, 94) = 3.234$, n.s.)．

回答変更率の区間ごとのステージ 2 の正答率を図 7 に示す．Correct 条件における回答変更率の区間でのステージ 2 の正答率に差があるかを検証するために，独立変数を回答率の各区間，従属変数をステージ 2 の正答率とする一要因分散分析を行った．その結果，主効果が認められた ($F(3, 78) = 64.97$, $p < .001$)．Bonferroni 法による多重比較では，0.4-0.6 と 0.6-0.8 の間以外の区間において有意差が認められた．次に Random 条件における回答変更率の区間でのステージ 2 の正答率に差があるかを検証するために，独立変数を回答率の各区間，従属変数をステージ 2 の正答率とする一要因分散分析を行った．その結果，主効果が認められた ($F(4, 94) = 4.717$, $p < .01$.)．Bonferroni 法による多重比較では，0.2-0.4 と 0.6-0.8 の間 ($p < .001$) に有意差が認められた．

4.3　回答変更率とテストタスクの成績

回答変更率の区間ごとの Post テストの正答率を図 8 に示す．Correct 条件における回答変更率の区間での Post テストの正

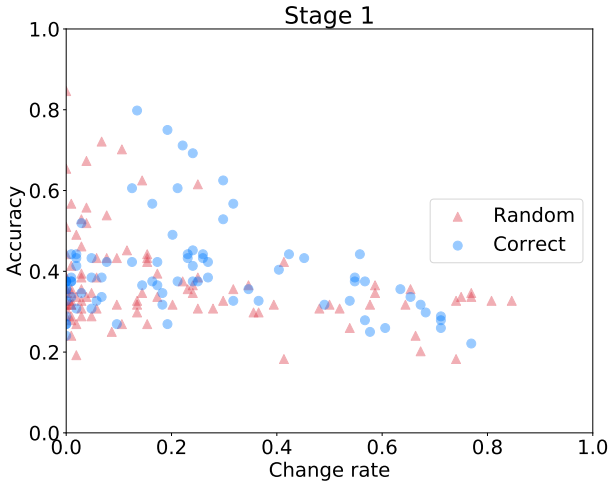

図 4　回答変更率と自己補正のステージ 1 の正答率

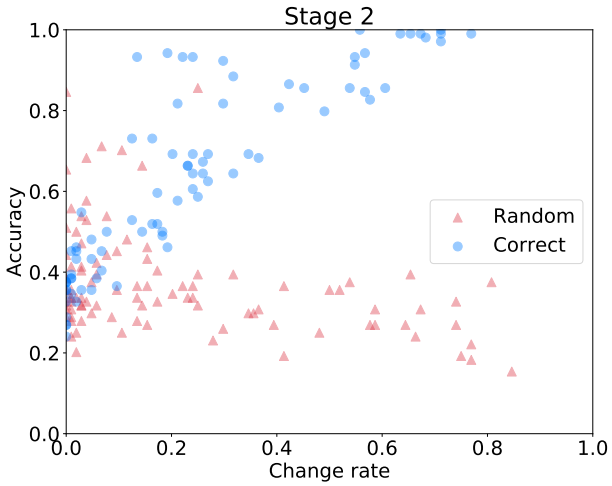

図 5　回答変更率と自己補正のステージ 2 の正答率

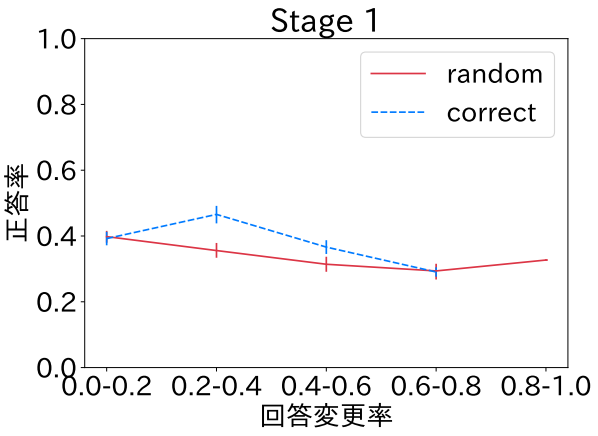

図 6　回答変更率の区間ごとの stage1 の正答率

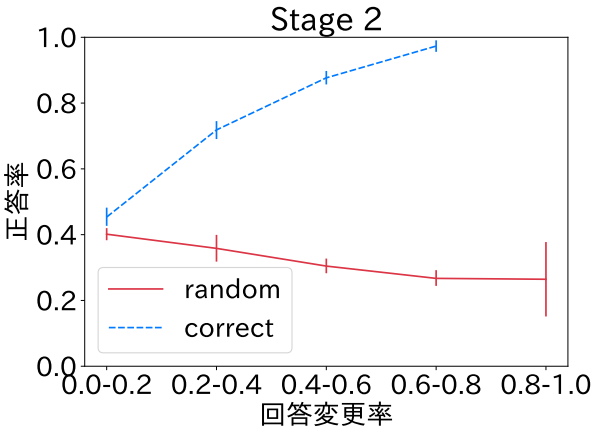

図 7　回答変更率の区間ごとの stage2 の正答率

答率に差があるかを検証するために，独立変数を回答率の各区間，従属変数を Post テストの正答率とする一要因分散分析を行った．その結果，主効果が認められた ($F(3, 78) = 5.701$, $p < .01$)．Bonferroni 法による多重比較では，0.0-0.2 と 0.2-0.4 の間 ($p < .01$)，0.2-0.4 と 0.6-0.8 の間 ($p < .05$) に有意差が認められた．

次に Random 条件における回答変更率の区間での Post テストの正答率に差があるかを検証するために，独立変数を回答率の各区間，従属変数を Post テストの正答率とする一要因分散分析を行った．その結果，主効果は認められなかった ($F(4, 94) = .343$, n.s.)．

回答変更率の区間ごとの Mid テストの正答率を図 9 に示す．Correct 条件における回答変更率の区間での Mid テストの正答率に差があるかを検証するために，独立変数を回答率の各区間，従属変数を Pre テストの正答率とする一要因分散分析を行った．その結果，主効果が認められた ($F(3, 78) = 4.001$, $p < .05$)．Bonferroni 法による多重比較では，0.0-0.2 と 0.2-0.4 の間 ($p < .05$)，0.2-0.4 と 0.6-0.8 の間 ($p < .05$) に有意差が認められた．Random 条件における回答変更率の区間での Mid テストの正答率に差があるかを検証するために，独立変数を回答率の各区間，従属変数を Pre テストの正答率とする一要因分散分析を行った．その結果，主効果は認められなかった ($F(4, 94) = 1.128$, n.s.)．

回答変更率の区間ごとの Pre テストの正答率を図 10 に示す．Correct 条件における回答変更率の区間での Pre テストの正答率に差があるかを検証するために，独立変数を回答率の各区間，従属変数を Pre テストの正答率とする一要因分散分析を行った．その結果，主効果の有意傾向が認められた ($F(3, 78) = 2.457$, $p < .1$)．Bonferroni 法による多重比較では，すべての組み合わせについて有意差が認められなかった．Random 条件における回答変更率の区間での Pre テストの正答率に差があるかを検証するために，独立変数を回答率の各区間，従属変数を Pre テストの正答率とする一要因分散分析を行った．その結果，主効果は認められなかった ($F(4, 94) = .94$, n.s.)．

4.4　回答変更率とワーカの成長度合い

各ワーカのテストタスクの正答率について，Post テストの正答率と Pre テストの正答率との差をワーカの成長度合いと

して考える．回答変更率の区間ごとの成長度合いの正答率を図 11 に示す．Correct 条件における回答変更率の各区間での成長度合いの平均値に差があるかを検証するために，独立変数を回答率の各区間，従属変数を成長度合いとする一要因分散分析を行った．その結果，主効果の有意差が認められた ($F(3, 78) = 3.325$, $p < .05$) ．Bonferroni 法による多重比較では，0.2-0.4 と 0.6-0.8 の間 ($p < .05$) にのみ有意差が認められた．Random 条件における回答変更率の各区間での成長度合いの平均値に差があるかを検証するために，独立変数を回答率の各区間，従属変数を成長度合いとする一要因分散分析を行った．その結果，主効果は認められなかった ($F(4, 94) = .234$, n.s.) ．

5 考　　察

5.1 回答変更率とテストタスクの正答率

複数の自己補正タスクの結果から算出した回答変更率と，各テスト時期のタスク結果の正答率の関係を調べた．その結果，Correct 条件の Mid テストと Post テストでは有意差が認められる回答変更率の区間が見られたが，Correct 条件の Pre テストおよび Random 条件のすべてのテスト時期では有意差が認められる区間は無かった．このとこから，テスト時期の正答率の改善は，参考回答が信頼できる場合において見られ，学習にはある程度の自己補正タスクの繰り返しが必要であると言える．正答率の改善は，回答変更率が 0.2〜0.4 のグループにのみ見られた．このことから，自己補正タスクで提示された参考回答を活用し，ワーカ自身の能力改善に繋げることが出来たのは回答変更率が特定の範囲に属している可能性が高いことが示唆された．Pre テストでは，回答変更率の区間毎の正答率に有意差が認められなかった．そのため単に事前テストからこのような正答率が改善するワーカを発見することは困難であると考えられる．しかし，適切な参考回答を提示する自己補正タスクを割り当てることで，ワーカの正答率の改善が期待できるため，彼らを発見し，適切な動機づけや，継続的に作業に取り組むことを促すことは重要である．本研究の実験では，自己補正の回答変更率は信頼できる可能性の高いワーカを発見する指標として活用できる可能性が示唆された．ただし，正答率の改善が見られるワーカが属する回答変更率の範囲は取り扱う課題により変動するものと考えられる．

5.2 回答変更率とワーカの成長度合い

複数の自己補正タスクの結果から算出した回答変更率と，ワーカの成長度合いの関係を調べた．その結果，Correct 条件において回答変更率が 0.2〜0.4 のワーカの成長度合いが高いことが分かった．この結果は，テストタスクの正答率での分析および考察を支持するものである．

5.3 回答変更率と自己補正タスクの正答率

複数の自己補正タスクの結果から算出した回答変更率と，自己補正タスクのステージ 1 の正答率の関係を調べた．その結果 Correct 条件では回答変更率の区間が 0.2〜0.4 のワーカは 0.6〜0.8 のワーカよりも正答率が高いことがわかった．一方で

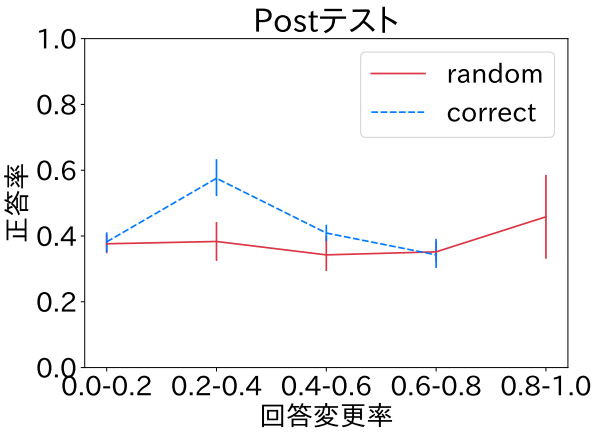

図 8　回答変更率の区間ごとの Post テストの正答率

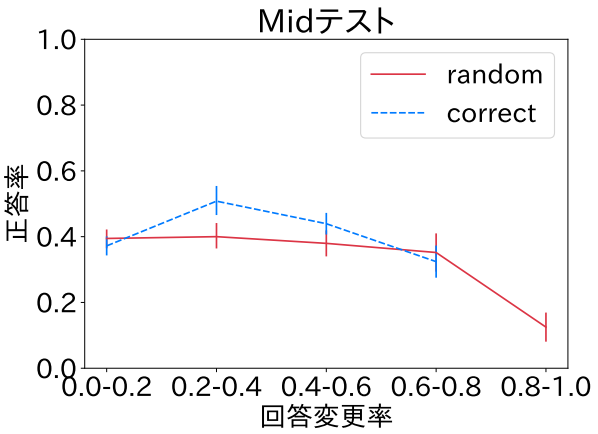

図 9　回答変更率の区間ごとの Mid テストの正答率

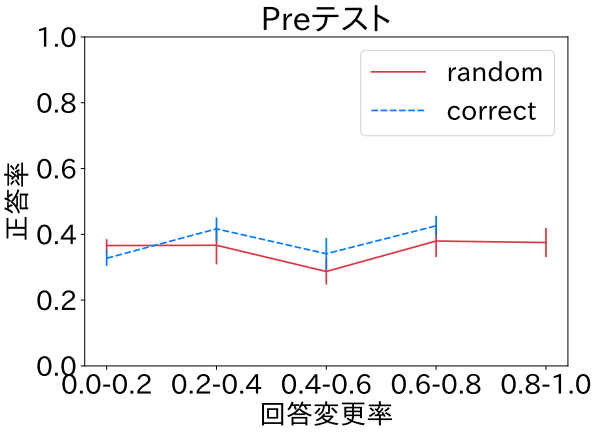

図 10　回答変更率の区間ごとの Pre テストの正答率

Random 条件ではいずれの区間においても有意差は見られなかった．Correct 条件のみで有意差が見られた理由としては，学習 1 と学習 2 のタスク結果を分けずに分散分析を行ったことが挙げられる．Correct 条件ではテスト時期が進むごとに，回答変更率の区間が 0.2〜0.4 のワーカについて正答率の改善が見

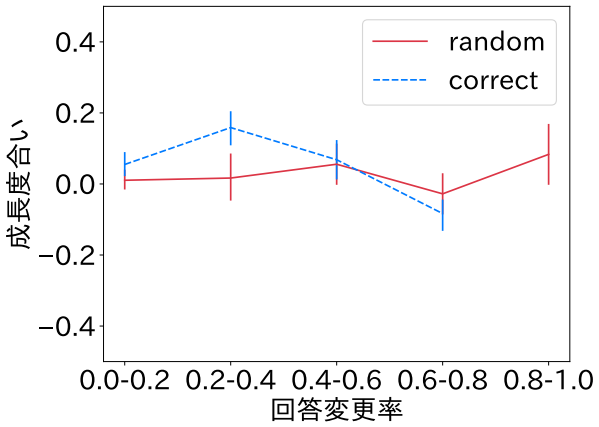

図 11　回答変更率の区間ごとの成長度合い

られた．そのため，学習が進んだワーカが自己補正タスクのステージ 1 に取り組んだ際にも，比較的信頼性の高い回答をもたらし，平均正答率の向上に繋がったということである．このことから，テストタスクの正答率が改善する様なワーカ，テストタスクだけでなく自己補正タスクのステージ 1 の正答率も改善することが示唆された．

同様に，回答変更率と自己補正タスクのステージ 2 の正答率の関係を調べた．その結果，Correct 条件では，回答変更率の各区間の多くに有意差が認められ，回答変更率が高くなるほど，正答率が参考回答の品質である 100%に近づく傾向が見られた．一方で，Random 条件では，回答変更率が 0.2〜0.4 と 0.6〜0.8 の間にのみ有意差が認められ，回答変更率が高くなるほど，正答率がタスク回答のチャンスレベルである 25&に近づく傾向が見られた．このことから，参考回答の条件によらず，回答変更率が高いワーカの自己補正タスクのタスク結果は参考回答を採用した可能性が高いということが示唆された．加えて，自己補正タスクにおいて回答を変更する場合，その多くは提示された参考回答を追従するのが大半であることが分かった．

6　まとめと今後の課題

本研究では，現実のクラウドワーカが自己補正を適用したマイクロタスクに取り組んだ場合の，回答の変更パターンの分析を試みた．実験結果は，複数の自己補正タスクにおける回答の変更頻度と，ワーカから得られるタスク結果の品質の関連性を示唆するものである．

回答の変更頻度が高いワーカは，回答の変更頻度が低いワーカと比べて，第一段階で選ぶ回答の品質が低い傾向が見られた．更に，自己補正タスクを繰り返すことで回答品質が改善するワーカには，回答の変更頻度が高すぎず，かつ低すぎない傾向が見られた．このことから，自己補正タスクにおける回答変更率を観察することで，信頼性の高いワーカを発見できる可能性が示唆された．

今後の課題として，本研究で注目した自己補正タスクにおけるワーカの回答パターンの特徴に基づいたワーカの分類や報酬設計の検討が挙げられる．ある時点で信頼できるワーカを見つけるだけでなく，今後信頼できるワーカへと成長する見込みがあるワーカを早期に発見し，そのようなワーカに対しても適切な動機づけを行うことが出来れば，クラウドソーシングにおけるワーカの拡充に貢献できる可能性がある．

一方で，マイクロタスクにおける回答変更率は，参考情報の信頼性だけでなく，取り扱う課題やタスク設計に左右されやすいことが考えられる．そのため，条件を変えた複数の実験を検討し，より一般的な傾向を捉える必要がある．

謝　辞

本研究の一部は JST CREST (#JPMJCR16E3) の支援による．タスクに参加いただいた Crowd4U のワーカの皆さんに感謝いたします．ワーカの一部リストは crowd4u.org にあります．

文　献

[1] Nihar Shah and Dengyong Zhou. No oops, you won't do it again: Mechanisms for self-correction in crowdsourcing. In Maria Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, Vol. 48 of *Proceedings of Machine Learning Research*, pp. 1–10, New York, New York, USA, 20–22 Jun 2016. PMLR.

[2] Masaki Kobayashi, Hiromi Morita, Masaki Matsubara, Nobuyuki Shimizu, and Atsuyuki Morishima. An empirical study on short- and long-term effects of self-correction in crowdsourced microtasks.

[3] Joseph Chee Chang, Saleema Amershi, and Ece Kamar. Revolt: Collaborative crowdsourcing for labeling machine learning datasets. In *Proceedings of the Conference on Human Factors in Computing Systems (CHI 2017)*. ACM Association for Computing Machinery, May 2017.

[4] Ryan Drapeau, Lydia B Chilton, Jonathan Bragg, and Daniel S Weld. Microtalk: Using argumentation to improve crowdsourcing accuracy. In *Fourth AAAI Conference on Human Computation and Crowdsourcing*, 2016.

[5] Steven Dow, Anand Kulkarni, Scott Klemmer, and Björn Hartmann. Shepherding the crowd yields better work. In *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work*, pp. 1013–1022. ACM, 2012.

[6] Ujwal Gadiraju, Besnik Fetahu, Ricardo Kawase, Patrick Siehndel, and Stefan Dietze. Using worker self-assessments for competence-based pre-selection in crowdsourcing microtasks. *ACM Transactions on Computer-Human Interaction (TOCHI)*, Vol. 24, No. 4, p. 30, 2017.

[7] Masayuki Ashikawa, Takahiro Kawamura, and Akihiko Ohsuga. Proposal of grade training method in private crowdsourcing system. In *Third AAAI Conference on Human Computation and Crowdsourcing*, 2015.

[8] Ryo Suzuki, Niloufar Salehi, Michelle S. Lam, Juan C. Marroquin, and Michael S. Bernstein. Atelier: Repurposing expert crowdsourcing tasks as micro-internships. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pp. 2645–2656, New York, NY, USA, 2016. ACM.

[9] Azad Abad, Moin Nabi, and Alessandro Moschitti. Autonomous crowdsourcing through human-machine collaborative learning. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '17, pp. 873–876, New York,

NY, USA, 2017. ACM.

[10] Edith Law, Ming Yin, Joslin Goh, Kevin Chen, Michael A. Terry, and Krzysztof Z. Gajos. Curiosity killed the cat, but makes crowdwork better. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pp. 4098–4110, New York, NY, USA, 2016. ACM.

[11] Nguyen Quoc Viet Hung, Duong Chi Thang, Matthias Weidlich, and Karl Aberer. Minimizing efforts in validating crowd answers. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, pp. 999–1014. ACM, 2015.

[12] Daniel Haas, Jason Ansel, Lydia Gu, and Adam Marcus. Argonaut: macrotask crowdsourcing for complex data processing. *Proceedings of the VLDB Endowment*, Vol. 8, No. 12, pp. 1642–1653, 2015.

[13] Ujwal Gadiraju, Ricardo Kawase, Stefan Dietze, and Gianluca Demartini. Understanding malicious behavior in crowdsourcing platforms: The case of online surveys. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pp. 1631–1640. ACM, 2015.

[14] Hyun Joon Jung and Matthew Lease. Modeling temporal crowd work quality with limited supervision. In *Third AAAI Conference on Human Computation and Crowdsourcing*, 2015.

[15] Manas Joglekar, Hector Garcia-Molina, and Aditya Parameswaran. Evaluating the crowd with confidence. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '13, pp. 686–694, New York, NY, USA, 2013. ACM.

[16] Akash Das Sarma, Aditya Parameswaran, and Jennifer Widom. Towards globally optimal crowdsourcing quality management: The uniform worker setting. In *Proceedings of the 2016 International Conference on Management of Data*, SIGMOD '16, pp. 47–62, New York, NY, USA, 2016. ACM.