人間+AIクラウドの相互作用による
クラウドソーシングの品質管理に関する研究

2022年03月

小林　正樹

人間+AIクラウドの相互作用による
クラウドソーシングの品質管理に関する研究

筑波大学
図書館情報メディア研究科
2022年03月

小林　正樹

人間+AI クラウドの相互作用による クラウドソーシングの品質管理に関する研究

概要

本論文では，人間による処理と計算機処理の協調による効率的な課題解決を実現するために，人間＋ AI クラウドにおけるクラウドソーシングの品質管理の問題に取り組んだ．

クラウドソーシングサービスを通して，人間のリソースや AI のリソースにアクセスすることで依頼者の課題解決する試みが普及している．クラウドソーシングとは，依頼者の課題を解決するために，インターネットを介して匿名で不特定多数の作業者 (人間ワーカ) に仕事 (タスク) を分散して依頼する枠組みである．代表的な活用例として，書籍等の電子化を行う reCAPTCHA や，ゲーム形式の作業を通してタンパク質の構造解析を行う Foldit が挙げられる．人間ワーカに対して画像や文章へのタグ付けや評価を依頼することで，現時点で計算機処理を行うためのアルゴリズム等が確立されていない処理を実現できる．このような情報処理の方式はヒューマンコンピュテーションと呼ばれ，これを活かすことで新しい問題に対して専用のシステムを構築するよりも素早く取り掛かることが可能である．一方で，クラウドソーシングサービスを通してソフトウェア開発を依頼する試みも行われている．特に，Kaggle に代表されるデータ解析プラットフォームでは，依頼者がデータや課題を公開し，参加者がその課題に対するデータ分析や予測モデルの開発に取り組む．各参加者が提案した手法の性能を競い合うことで，依頼者の想像を超えた成果を得られる可能性がある．このように，クラウドソーシングにより人間や AI のリソースにアクセスすることで，依頼者の知識やスキルにスキルに囚われることなく，多様なアプローチで依頼者の課題に取り組める．

しかしながら，インターネットを通してアクセスできる人間や AI のリソースをどのように組み合わせれば，依頼者の課題を効率的に解決できるかは自明でない．マイクロタスク型のクラウドソーシングサービスを利用して大量の画像にラベル付けを行いたいとしても，ある時点で依頼可能なワーカの処理能力の制限によって全ての処理を達成できない可能性がある．一方，大量の画像を処理できる機械学習モデルやアルゴリズムの開発を依頼し，成果物を入手したとしても，十分な品質で処理できる保証はない．そのため，人間のリソースと AI のリソースの組み合わせ方を設計することが重要である．特定の課題を解決するために，利用可能なリソースを適切に活用してシステムを構築すれば，合理的なコスト・品質での処理の実現が期待できる．リソースの合理的な組み合わせ方は依頼者の判断に委ねられるが，適切な組み合わせは対象となる課題や処理結果に求められる品質に左右される．性能の優れた機械学習モデルを効率的に構築するという目的においては，

Human-in-the-loop 機械学習の文脈での研究がある．これは機械学習モデルの出力を評価しながら，次に学習データとして必要なデータを選択し，人間のワーカが順次アノテーションするという一連の処理を繰り返すものである．これにより，少ない人間ワーカの労力でより優れた機械学習モデルを入手することが期待できる．このようなアプローチでは，成果物として得られた AI による処理を採用する，もしくは採用しないという意思決定を依頼者が行う必要がある．さらに，Human-in-the-loop 機械学習ではループの実行前に，対象とする機械学習モデルを決定する必要があり，複数の候補があったとしても途中で対象の機械学習モデルを切り替えることはできない．このようなアプローチによる人間と AI のリソースの統合は，依頼者による意思決定を課題ごとに行う必要があるため，スケーラブルであるとは言えない．

そこで本論文では，不特定多数の人間ワーカおよび AI のワーカで構成される人間+AI クラウドにおける効率的なタスク処理の実現に向けた，タスク結果の品質管理の問題に取り組んだ．品質やアルゴリズムの仮定が置けない人間ワーカおよび AI ワーカが利用可能な状況において，タスク処理を通じた両者の相互作用により，依頼者が求める品質水準でタスクを処理することを目指す．研究目的を達成するため，次の 3 つの研究に取り組む．(研究 1) 人間+AI クラウドへのタスク割り当て：タスク集合と要求精度，AI ワーカの集合が与えられた時に，依頼者の要求精度を満たすように，各ワーカへタスクを割り当てることを理論的に保証するアルゴリズムを開発する．AI ワーカにタスクを割り当てるかどうかの評価は，正しい回答を返すと仮定した人間ワーカのタスク結果を用いて行う．(研究 2) 人間ワーカからの誤答を考慮したタスク割り当て手法の改良：研究 1 で提案したタスク割り当てアルゴリズムと多数決などのタスク結果の集約手法を組み合わせることで，人間ワーカの誤答を考慮することができるが，人間ワーカへのより多くの割り当てが必要である．そこで，人間ワーカと AI ワーカのタスク結果の不一致に着目し，それに基づいて人間ワーカに対して追加タスクを割り当てることで，品質管理が必要なタスクを削減しながら人間ワーカおよび AI ワーカにタスクを割り当てる手法を提案する．(研究 3) 人間ワーカからより正確なタスク結果を引き出すタスク設計：研究 2 では，多数決による人間ワーカからのタスク結果品質の向上と追加割り当てによるコストのトレードオフを制御する手法を検討した．しかし，多数決に基づいた品質管理は，人間ワーカの過半数が不正確な回答をもたらす状況ではうまく機能しない．そこで，個々の人間ワーカからより正確なタスク結果を得るためのタスク設計手法を検討する．

本論文は 5 章で構成される．第 1 章では，研究の背景と目的，研究方法，本研究の貢献，論文構成について述べた．第 2 章では，本研究の関連研究について整理した．

第 3 章では，人間＋ AI クラウドにおける自動的なタスク割り当てについて述べ

た．リクエスタが設定した要求精度を満たすように人間ワーカおよびAIワーカに対してタスクを割り当てることを目的として，自動的な割り当てを実現するために，人間＋AIクラウドタスク割り当て問題（Human+AI Crowd Task Assignment Problem, HACTAP）を定義した．リクエスタの要求精度を満たしながら，AIワーカへのタスク割り当て数を最大化するために，AIワーカから得られるタスク結果の全体を評価する代わりに，タスク結果のラベルが同じであるようなタスクの部分集合（タスククラスタ）毎にAIワーカの統計的な評価を行う，Clusterwise Test-based Assignment（CTA）を提案した．さらに，CTAが各タスククラスタを独立に評価するのに対し，すでに人間ワーカおよびAIワーカに割り当て済みのタスクおよび次の評価対象となるタスククラスタを考慮した全体的なタスク結果品質を評価するGlobal Test-based Assignment（GTA）を提案した．CTAおよびGTAが少なくとも要求精度を満たす割り当てを求められることについて理論解析を行った．ベンチマークデータセットおよび水害被害判定タスクを用いて，単一のAIワーカの全体的な性能を評価するベースライン手法，および単一のAIワーカを対象とした能動学習を行うベースライン手法 (Active Learning-based Assignment, ALA) と提案手法の比較実験を行った．

第4章ではまず，人間ワーカから得られるタスク結果品質が不正確な状況におけるCTAの振る舞いを分析した．HACTAPでは人間ワーカのタスク結果は単なる成果物としてだけでなく，AIワーカの学習と評価に利用される．したがって，人間ワーカから得られるタスク結果品質は特に重要であるが，現実のクラウドソーシングでは，様々な理由で人間ワーカからのタスク結果が不正確な可能性があり，これにより最終的なタスク結果品質の低下を引き起こす．そのため，多数決などの集約手法を適用することが一般的であるが，このコストをできるだけ削減したい．本章では，人間ワーカとAIワーカの回答の不一致に着目し，人間ワーカの追加タスクの必要性を判定することで，人間ワーカのタスク数の増加を抑えながらタスク結果品質を改善する手法を提案する．ベンチマークデータセットを用いた実験結果から，不確実な人間ワーカとAIワーカが相互にタスク結果を共有することで，タスク結果品質を改善しながら効率的にタスクを処理する仕組みが構築可能であることを示す．

第5章では，Human ＋ AIクラウドにおける人間ワーカから得られるタスク結果の品質管理について議論した．第4章で述べたように，人間ワーカから得られるタスク結果を用いてAIワーカの学習および評価を行うため，人間ワーカのタスク結果品質はHuman+AIクラウドから得られるタスク結果品質を左右する重要な要素である．多数決に代表されるタスク結果の集約手法を組み合わせることで，集約結果として得られるタスク結果品質を改善することに繋がるが，個々の人間ワーカから得られるタスク結果品質を向上させることがより重要である．クラウドソーシン

グの分野において様々な研究がこの課題に取り組んでおり，本章ではその１つである Shah らが提案した自己補正に着目した．自己補正はタスク結果の品質を改善することを目的としたタスク設計手法であり，１つのタスクに２度の回答の機会を与えることで，ワーカ自身が自分の誤答を補正できるのが特徴である．自己補正は，多数決をはじめとするタスク結果の集約手法や，優れたワーカの選出，タスク割り当て手法などと組み合わせることが容易であることから，多くの場面で活用できる可能性がある．しかし，自己補正の提案論文では，タスク結果の品質改善についてのシミュレーションによる評価のみが行われており，現実のクラウドソーシングにおいてもシミュレーションと同様の効果が得られるかは不明であった．そこで，現実のクラウドワーカが自己補正を適用したタスクに取り組む実験により，自己補正がタスク結果やワーカにもたらす効果を検証した．

第 6 章では，第一章で述べた研究目的を踏まえて，各章で取り組んだ研究とそこから得られた結果について議論した．

A Study on Quality Control of Crowdsourcing by Interaction of the Human+AI Crowd

Abstract

This study addresses the quality control of crowdsourcing in human + AI crowds to realize efficient problem-solving through cooperative computation between humans and AIs.

Crowdsourcing is widely used for solving requesters ' problems by accessing human and AI resources. Crowdsourcing is an Internet-based approach that distributes work (tasks) to an unspecified number of human workers anonymously to solve problems. Typical examples include reCAPTCHA, which digitizes paper-based documents, and Foldit, which analyzes the structure of proteins through game-style tasks. Crowdsourcing realizes processes for which computer-processing algorithms have not yet been established by assigning tasks to human workers. This type of information-processing approach is termed human computation. Utilizing this approach makes it possible to address problems more quickly than building a specific system for new issues. Conversely, some interesting projects handle software development with human developers through crowdsourcing. In particular, a data analysis platform named Kaggle allows requesters to publish their data and the problems. The participants then work on data analysis and develop prediction models for the problems. Using such projects, the requesters may receive unexpectedly excellent outcomes by competing for the solutions proposed by each participant. By accessing human and AI resources through crowdsourcing platforms, the requester's problems can be addressed by employing various approaches that exceed the requester's knowledge and skills.

However, it is not trivial how anonymous human and AI resources accessed through the Internet can be combined optimally to solve the requesters' problems. For example, if requesters consider microtask crowdsourcing to have labels for numerous images, it may be impossible because the processing capacity of human workers is limited. Conversely, there is another approach to outsource software development of machine learning or algorithms to label the images; however, there is no guarantee of the quality of output from the software. Therefore, it is critical to design methods that combine human and AI resources. A properly designed system works well at a reasonable cost and quality while considering the available resources. Requesters must find a suitable combination of resources depending on the target

problem and the required quality. Research has been conducted in the context of human-in-the-loop machine learning efficiently to train high-performance machine learning models. Humans-in-the-loop ML aims to train a better ML model with less human worker effort by finding the required data items for the next training iteration and assigning such data items to human workers. Nevertheless, the requesters must decide whether to accept the ML model because having a human-in-the-loop does not guarantee the ML model's performance. Furthermore, in human-in-the-loop ML, the requesters must select an ML model before executing the loop, and, once in use, it cannot be changed, even if other candidates are available. Such an approach is not scalable because the requester has many parameters to be determined.

This study addresses quality control to realize efficient task processing with human workers and AI workers in the crowd. In situations where human workers and AI workers who cannot make assumptions about the quality are used, we aim to process tasks at the requester ' s required quality level by the interaction of them during task processing. The following three aspects are studied to achieve the research goal: (1) Task assignment to the human + AI crowd: A novel human+AI crowd task assignment problem (HACTAP) is defined that calculates a set of task assignments for human and AI workers to the tasks. Then some theoretical guarantees are provided for the proposed algorithms to the HACTAP. (2) Improve the quality of the results obtained from human workers in the HACTAP: In real-world crowdsourcing, the task results from human workers are sometimes noisy. This study applies task result aggregation methods to the proposed task assignment algorithms. In addition, we tried to reduce the number of task assignments to human workers by finding disagreement between human and AI workers. Our method improves the noisy task results from each human worker while reducing human worker assignments. (3) Task design to derive more reliable task results from individual human workers: In study (2), we controlled the quality of task results by aggregation methods for task results like majority vote. However, quality control does not work well when most human workers produce inaccurate results. Therefore, we consider task design methods to obtain more accurate task results from individual human workers by giving opportunities to improve task results.

This thesis consists of six chapters. Chapter 1 describes the background and purpose, the method, contributions, and structure of this thesis. Chapter 2 summarizes the related work.

Chapter 3 describes the method of automatic task assignment in the human + AI crowd. We define the human + AI crowd task assignment problem (HACTAP),

which aims to calculate task assignment to human and AI workers while satisfying the requester ‘ s quality requirement. To maximize the number of tasks assigned to the AI workers, each subset of tasks (task cluster) that contains the same output type from the AI workers is evaluated instead of the AI workers ’ overall performance (Clusterwise Test-based Assignment, CTA). We also propose GTA (Global Test-based Assignment) that comprehensively evaluates the quality of task results by considering the overall quality that has already been assigned to both human and AI workers and the next task cluster candidate. Then, we provide theoretical analysis for the proposed CTA and GTA to guarantee that the overall accuracy satisfies the requirement. We conducted comparative experiments with both open benchmark and real-world datasets to evaluate the performance of CTA and GTA with two baselines.

In Chapter 4, we analyze the performance of CTA in situations where the accuracy of human workers is not reliable. In the human + AI crowd situation, AI workers use task results from human workers for training and evaluation. Therefore, the quality of task results from human workers directly affects the performance of AI workers and the overall task result quality. Applying an aggregation method, such as majority voting, is common, but it takes more human effort. This chapter describes the method for reducing the number of assignments to human workers by evaluating the importance of quality control for each task by comparing the results from human and AI workers. Experimental results showed that the proposed method reduces the effort required by human workers by finding tasks that require majority voting than overall majority voting.

Chapter 5 discusses quality control for task results obtained from human workers in the Human + AI crowd. As we mentioned in chapter 4, the quality task result from human workers affects the performance of AI workers and overall quality, and task result aggregation methods are commonly used. However, it is crucial to improve the quality of task results obtained from individual human workers. Various crowdsourcing studies address this problem, and this chapter focuses on one of them, the self-correction proposed by Shah et al. Self-correction is a task design method aimed to improve the quality of task results, and it provides an opportunity to fix the final task result. Self-correction can be easily combined with other methods such as majority voting, finding good workers, and task assignment methods. However, in the proposed paper on self-correction, the authors evaluated the effect of self-correction by simulation only. Therefore, we verified the effect of self-correction on the quality of task results and observed the behavior of actual crowd workers

working on self-correction tasks in real-world crowdsourcing.

Finally, Chapter 6 reviews the research objectives described in Chapter 1 and discusses the research outcomes for each chapter.

目 次

図 目 次

第1章　序論

1.1　研究背景

本論文では，人間による情報処理と計算機処理の協調による効率的な課題解決を実現するために，人間＋ AI クラウドにおけるクラウドソーシングの品質管理の問題に取り組んだ．

クラウドソーシングサービスを通して，人間のリソースや AI のリソースにアクセスすることで依頼者の課題解決する試みが普及している．クラウドソーシング [1] は，依頼者の課題を解決するために，インターネットを介して匿名で不特定多数の作業者 (人間ワーカ) に仕事 (タスク) を分散して依頼する枠組みである（図 1.1）．代表的な活用例として，紙媒体の書類等の電子化を行う reCAPTCHA[2] や，ゲーム形式の作業を通してタンパク質の構造解析を行う Foldit[3] が挙げられる．人間ワーカに対して画像や文章へのタグ付けや評価を依頼することで，現時点で計算機処理を行うためのアルゴリズム等が確立されていない処理を実現できる．このような情報処理の方法はヒューマンコンピュテーション [4] と呼ばれ，これを活かすことで新しい問題に対して専用のシステムを構築するよりも素早く課題に取り掛かることが可能である．代表的な商用クラウドソーシングプラットフォームとして Amazon Mechanical Turk[1]や Appen（旧 Figure Eight および Crowdflower）[2]が挙げられ，日本国内においては Yahoo!クラウドソーシング[3]が挙げられる．これらは画像へのラベル付のような数秒から数分で作業を完了することができる，マイクロタスク型の仕事を依頼することを主として設計されているが，マイクロタスクに限定されない仕事を依頼することができるプラットフォームとして Upwork[4]やクラウドワークス[5]，ランサーズ[6]が挙げられる．

一方で，クラウドソーシングサービスを通してソフトウェア開発を依頼する試みも行われている．近年，コンペティション形式でアルゴリズムや機械学習モデルの開発をクラウドソースする Kaggle[7]や AICrowd[8]，Topcoder[9]など，ボランティアに

[1] https://www.mturk.com/
[2] https://appen.com/
[3] https://crowdsourcing.yahoo.co.jp/
[4] https://www.upwork.com/
[5] https://crowdworks.jp/
[6] https://www.lancers.jp/
[7] https://www.kaggle.com/
[8] https://www.aicrowd.com/
[9] https://www.topcoder.com/

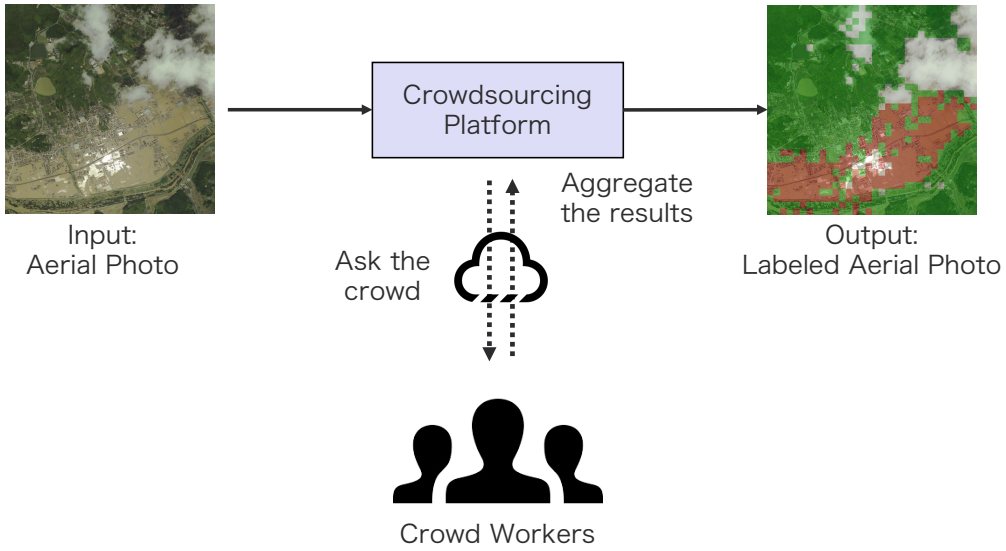


図 1.1: クラウドソーシングの概要図：図の例では，航空写真をクラウドソーシングプラットフォームに入力すると，プラットフォームが作業を細分化してクラウドワーカに依頼し，集めた結果を集約して最終的なラベル済み航空写真を得る．

よるソフトウェア開発が普及しつつある [5]．特に，Kaggle に代表されるデータ解析プラットフォームでは，依頼者がデータや課題を公開し，参加者がその課題に対するデータ分析や予測モデルの開発に取り組む．各参加者が提案した手法の性能を競い合うことで，依頼者の想像を超えた成果を得られる可能性を秘めている．このように，クラウドソーシングプラットフォームを通して人間や AI のリソースにアクセスすることで，依頼者の知識やスキルに制限されることなく，多様なアプローチで課題に取り組むことが可能である．

1.2 研究課題

クラウドソーシングで得られるのは外部協力者から得られた成果物である以上，それらの品質や性能が十分であるかを保障することは難しく，依頼者の判断に委ねられる．このような性能や能力が仮定できない不特定多数のワーカのリソースをどのようにして適切に活用するかが課題である．しかしながら，インターネットを通してアクセスできる人間や AI のリソースをどのように組み合わせれば，依頼者の課題を効率的に解決できるかは自明ではない．

マイクロタスク型のクラウドソーシングサービスを利用して大量の画像にラベル付けを行いたいとしても，ある時点で依頼可能なワーカの処理能力の制限によって全ての処理を達成できない可能性がある．作業可能なワーカが限られていることは，クラウドソーシングのスケーラビリティを左右する要因の１つである．

一方，大量の画像を処理できる機械学習モデルやアルゴリズムの開発を依頼し成

果物を入手したとしても，十分な品質で処理が可能である保証はない．性能を高めるためには大量かつ高品質のラベル済みデータを訓練データとしてそのモデルに学習させる必要があるが，ラベル済みデータの構築には人手による作業を伴うため，大規模なデータセットの構築には膨大なコストがかかる．そこで，人間のリソースと AI のリソースの組み合わせ方を設計することが重要である．特定の課題を解決するために，利用可能なリソースを適切に設計してシステムを構築すれば，合理的なコスト・品質での処理を実現できる．リソースの合理的な組み合わせ方は依頼者の判断に委ねられるが，最適な組み合わせ方は対象となる課題や処理結果に求められる品質により異なると考えられる．

性能の優れた機械学習モデルを効率的に学習するという目的においては，Human-in-the-loop ML という文脈で研究がなされている．これは機械学習モデルの出力を評価しながら，次に学習データとして必要なデータに人間のワーカが順次アノテーションしていく処理を繰り返すものである．これにより，少ない人間ワーカの労力でより優れた機械学習モデルが得ることを目指すものである．

AI ワーカを入手した後に，それらを評価して問題解決に適用するまでのプロセスは通常，依頼者が手動で行う．ただし，必ず AI が活用できるとも限らない．例えば，100 万件のデータに対してラベルを入手したい場合に，1 万件のラベル付けをクラウドソースしてコンペティションを行い，性能が最も優れたモデルを採用しても，所定の品質に到達しない場合がある．この場合，追加のラベル付けを行った上で改めてコンペティションを実施したり，全てのタスク処理を人間に依頼するなどの判断が必要である．このような意思決定プロセスはスケールせず，大量の AI ワーカを利用可能な状況で，人間ワーカとの適切な分担を見つけるのは困難である．前述の能動学習や，複数の機械学習モデルを組み合わせるアンサンブル学習などを組み合わせることでより効率的なワークフローを設計できる可能性はあるものの，より深い技術や知識が求められるにも関わらず，期待する性能でのデータ処理を実現できる保証はない．

1.3 本研究の目的

本研究の目的は，品質やアルゴリズムの仮定が置けない人間ワーカおよび AI ワーカが利用可能な状況において，タスク処理を通じた両者の相互作用により，依頼者が求める品質水準でタスクを処理する自動的な枠組みを実現することである．

大量のデータを処理するにあたり，人間ワーカのみがタスクを処理するのでは処理性能の限界がある．一方で，外部協力者から得られた不特定多数の機械学習モデル等のソフトウェアをどのように活用するのが適切であるかは自明ではなく，これには依頼者自身による試行錯誤のプロセスを伴う．そこで本研究では，AI をワーカ

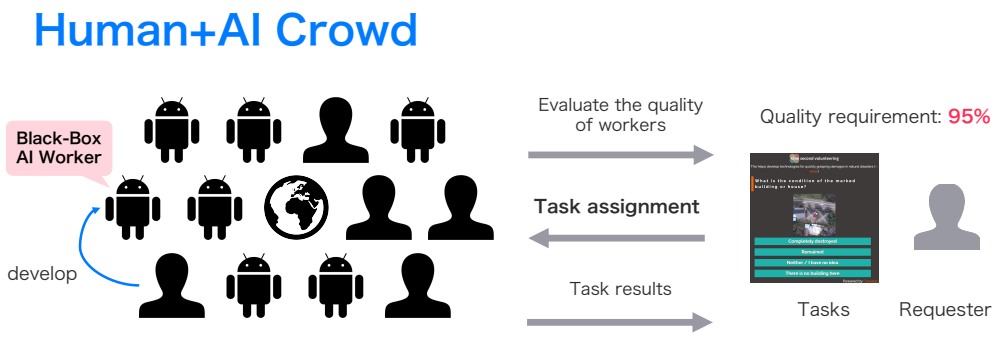


図 1.2: 人間+AI クラウドの概要図

としてモデル化（AI ワーカ）し，人間ワーカと同様にタスクを依頼することを検討する．公募によって得られた不特定多数の人間ワーカおよび AI ワーカで構成される群衆のことを**人間+AI クラウド**と呼ぶ（図 1.2）．

品質やアルゴリズムの仮定が置けない人間ワーカおよび AI ワーカが利用可能な状況において，タスク処理を通じた両者の相互作用により，依頼者が求める品質水準でタスクを処理することを目指す．人間+AI クラウドを活用し，大量のデータに対して効率的で高品質なタスク処理を実現するにあたり検討すべき課題は多いが，本研究では特に重要な次の 3 つに着目する．

（研究 1）人間＋ AI クラウドにおけるタスク結果品質を考慮した動的なタスク割り当て タスク集合と要求精度，現時点で利用可能な AI ワーカの集合が与えられた時に，依頼者の要求精度を考慮しながら，より多くのタスクを AI ワーカに割り当てるように，人間ワーカおよび AI ワーカへのタスク割り当てを決定する問題に取り組む．より多くのタスクを AI ワーカに割り当てながら，タスク割り当ての結果から得られる成果物の品質が少なくとも要求精度を上回ることを理論的に保証するアルゴリズムを開発する．AI ワーカの学習と評価には，その時点で利用可能な人間ワーカのタスク結果を用いる．

（研究 2）人間+AI クラウドの相互作用に基づくタスク割り当て手法の拡張 人間ワーカのタスク結果は単なる成果物としてだけでなく，AI ワーカの学習と評価に利用される．したがって，人間ワーカから得られるタスク結果品質は特に重要だが，現実のクラウドソーシングでは，様々な理由で人間ワーカからのタスク結果が不正確な可能性がある．そのため，タスク結果の信頼性を高めるために，多数決などの品質管理手法を適用することが一般的であるが，大量のデータを処理する状況ではコストが膨大になるため，このコストを削減したい．そこで，人間ワーカと別のワーカのタスク結果の不一致に着目し，必要なデータにのみ追加タスク等を生成することで，人間ワーカから得られるタスク結果の品質向

上を図る．これにより，人間ワーカのタスク結果から学習する種類の AI ワーカの性能が向上し，また研究 1 の割り当て手法の性能改善に繋がる．

（研究 3）人間ワーカからより正確なタスク結果を引き出すためのタスク設計 研究 2 では，多数決による人間ワーカから得られるタスク結果品質の向上と多数決のための追加タスク生成のコストのトレードオフを制御する方法を検討した．ただし，多数決に基づいた品質管理は，人間ワーカの過半数が不正確な回答をもたらす状況ではうまく機能しない．そこで，個々の人間ワーカからより正確なタスク結果を得るためのタスク設計手法が重要である．

1.4 本研究の貢献

本論文の貢献は次の通りである．

- 人間＋ AI クラウドの各ワーカへのタスク割り当ての問題（HACTAP）を定式化した．この問題に対して，依頼者の要求精度を満たすようなタスク割り当てを求めるアルゴリズムを提案し，アルゴリズムが決定したタスク割り当てで得られるタスク結果品質が要求精度を満たすことを理論解析した．
- 人間ワーカから得られるタスク結果が不正確な条件では，研究 1 で提案するタスク割り当てアルゴリズムが要求精度を満たす割り当てを求められないこと，人間ワーカへのタスク割り当てに多数決を適用することでタスク結果品質を向上できることを実験的に示した．さらに，人間ワーカと AI ワーカのタスク結果の不一致に基づいて，多数決による品質管理が必要なタスクを選択することで，タスク結果品質を低下させることなく，人間ワーカへのタスク割り当て数を削減できることを示した．
- 人間ワーカのタスク結果品質を改善するためのタスク設計手法である自己補正タスクについて，現実の人間ワーカによる実験を行い，その有効性を示した．さらに，人間ワーカが自己補正タスクに繰り返し取り組むことで，人間ワーカのスキル向上に繋がることを実験的に示した．

1.5 論文の構成

本論文は 5 章で構成される．第 1 章では，研究の背景と目的，研究方法，本研究の貢献，論文構成について述べた．第 2 章では，本研究の関連研究について整理した．

第 3 章では，人間＋ AI クラウドにおける自動的なタスク割り当てについて述べた．リクエスタが設定した要求精度を満たすように人間ワーカおよび AI ワーカ

に対してタスクを割り当てることを目的として，自動的な割り当てを実現するために，人間＋ AI クラウドタスク割り当て問題（Human+AI Crowd Task Assignment Problem, HACTAP）を定義した．リクエスタの要求精度を満たしながら，AI ワーカへのタスク割り当て数を最大化するために，AI ワーカから得られるタスク結果の全体を評価する代わりに，タスク結果のラベルが同じであるようなタスクの部分集合（タスククラスタ）毎に AI ワーカの統計的な評価を行う，Clusterwise Test-based Assignment（CTA）を提案した．さらに，CTA が各タスククラスタを独立に評価するのに対し，すでに人間ワーカおよび AI ワーカに割り当て済みのタスクおよび次の評価対象となるタスククラスタを考慮した全体的なタスク結果品質を評価する Global Test-based Assignment（GTA）を提案した．CTA および GTA が少なくとも要求精度を満たす割り当てを求められることについて理論解析を行った．ベンチマークデータセットおよび水害被害判定タスクを用いて，単一の AI ワーカの全体的な性能を評価するベースライン手法，および単一の AI ワーカを対象とした能動学習を行うベースライン手法 (Active Learning-based Assignment, ALA) と提案手法の比較実験を行った．

第 4 章ではまず，人間ワーカから得られるタスク結果品質が不正確な状況における CTA の振る舞いを分析した．HACTAP では人間ワーカのタスク結果は単なる成果物としてだけでなく，AI ワーカの学習と評価に利用される．したがって，人間ワーカから得られるタスク結果品質は特に重要であるが，現実のクラウドソーシングでは，様々な理由で人間ワーカからのタスク結果が不正確な可能性があり，これにより最終的なタスク結果品質の低下を引き起こす．そのため，多数決などの集約手法を適用することが一般的であるが，このコストをできるだけ削減したい．本章では，人間ワーカと AI ワーカの回答の不一致に着目し，人間ワーカの追加タスクの必要性を判定することで，人間ワーカのタスク数の増加を抑えながらタスク結果品質を改善する手法を提案する．ベンチマークデータセットを用いた実験結果から，不確実な人間ワーカと AI ワーカが相互にタスク結果を共有することで，タスク結果品質を改善しながら効率的にタスクを処理する仕組みが構築可能であることを示す．

第 5 章では，Human ＋ AI クラウドにおける人間ワーカから得られるタスク結果の品質管理について議論した．第 4 章で述べたように，人間ワーカから得られるタスク結果を用いて AI ワーカの学習および評価を行うため，人間ワーカのタスク結果品質は Human+AI クラウドから得られるタスク結果品質を左右する重要な要素である．多数決に代表されるタスク結果の集約手法を組み合わせることで，集約結果として得られるタスク結果品質を改善することに繋がるが，個々の人間ワーカから得られるタスク結果品質を向上させることがより重要である．クラウドソーシングの分野において様々な研究がこの課題に取り組んでおり，本章ではその１つで

ある Shah らが提案した自己補正に着目した．自己補正はタスク結果の品質を改善することを目的としたタスク設計手法であり，１つのタスクに２度の回答の機会を与えることで，ワーカ自身が自分の誤答を補正できるのが特徴である．自己補正は，多数決をはじめとするタスク結果の集約手法や，優れたワーカの選出，タスク割り当て手法などと組み合わせることが容易であることから，多くの場面で活用できる可能性がある．しかし，自己補正の提案論文では，タスク結果の品質改善についてのシミュレーションによる評価のみが行われており，現実のクラウドソーシングにおいてもシミュレーションと同様の効果が得られるかは不明であった．そこで，現実のクラウドワーカが自己補正を適用したタスクに取り組む実験により，自己補正がタスク結果やワーカにもたらす効果を検証した．

第 6 章では，第一章で述べた研究目的を踏まえて，各章で取り組んだ研究とそこから得られた結果について議論した．

第2章　関連研究

2.1　クラウドソーシングにおける品質管理

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は重要な研究課題の１つであり，これまでに様々な研究がこの問題に取り組んできた [6, 7]．不特定多数の人間ワーカからより正確な成果物を得るための様々な手法が提案されており，典型的な手法には複数のワーカの作業結果の統合，品質の高いワーカの検出 [8][9]，ワーカを訓練するためのタスクの導入，タスク設計の改良 [10]，報酬設計 [11] [12] などが挙げられる．

タスク結果の品質管理のために，本研究と既存手法を組み合わせることは有効な手段である．本研究が提案する割り当てアルゴリズムは人間ワーカのタスク結果品質を多数決により改善できることを前提とするものである．一方で，大半のワーカが誤った回答をするような，多数決が機能しない状況に対応するために，各ワーカの能力や対象のタスク以外への結果を考慮した回答統合手法と提案アルゴリズムを組み合わせることは興味深い課題である [13, 14, 15, 16]．Shah らの自己補正も既存のタスク結果の品質管理手法と統合が可能である．

2.2　能動学習

人間によりラベル付けされたデータに基づいて機械学習モデルを学習する際に，機械学習モデルの出力により次に人間によるラベル付けを必要とするデータを決定して，より少ないラベル付データで高性能なモデルを学習できることが知られている．このような方式は能動学習と呼ばれ，コストの制約がある，複数の性能の異なる人間ワーカに問い合わせが可能であるといった状況における問い合わせ戦略が研究されている [17, 18]．能動学習の目的は，与えられた予算で機械学習モデルの性能を最大化することであるため [19, 20]，タスク数が固定である問題に対しては有効であるとは限らない [21]．

能動学習と本研究が取り組む問題は次の２つの観点から異なる．(1) 能動学習の目的は予算内で対象の機械学習モデルの性能の最大化であるのに対し，本研究では AI ワーカと人間ワーカに適切にタスクを割り当てることで，全体として要求精度を満たすのが目的である．(2) 能動学習は学習ループの実行を開始する前に，対象の

機械学習モデルを与える必要があるが，本研究ではタスク処理の進行中に，ブラックボックスである複数の AI ワーカの増減を許す.

2.3 複数の機械学習モデルの統合

複数の機械学習モデルが利用可能な状況では，それらの出力を統合（モデルアンサンブル）し，より精度の高いモデルを構築できることが知られている [22][23]．しかし，機械学習モデルの候補やどのように組み合わせることが精度向上に繋がるかは自明ではなく，モデル設計者による実験が求められる．アンサンブル学習により得られた一連の AI ワーカを単一の AI ワーカに統合することで，“all-or-nothing” アプローチで AI ワーカの評価が可能である．ただし，動的なタスク割り当てを行う HACTAP では，より早い段階で AI ワーカに簡単なタスクを割り当て，人間ワーカは AI ワーカによって困難なタスクを集中して割り当てる必要がある．この問題に対処するために，本研究のタスククラスタに基づく割り当て戦略は有効である．

2.4 人と計算機の協調による課題解決

実世界の課題を解決するために人と計算機を統合する様々な研究がなされており [24, 25, 26, 27, 28]，計算機の活用することで人間の労力を削減することができる [29]．典型的なアプローチは，AI を利用して人間ワーカの作業が必要なデータを選択することである [30, 31, 32, 33]．分類モデルにおいて，分類精度の低下を避けるために，実際のラベルの代わりに拒否ラベルを返す特別な ML モデルを構築することも，人間ワーカによる作業が必要なデータを選択するために有効である [34]．AI の出力を用いて作業者を含む人間の学習や訓練に生かすことも，人と計算機の融合の重要なアプリケーションの１つである [35, 36, 37].

クラウドソーシングにより得られた分類タスクの結果を，依頼者とのインタラクションを通して動的にクラスタリングする研究がある [38]．分類のためのクラスタ数やラベルの候補を事前に決めるのではなく，データの性質に応じて適切なクラスタを生成することができる．このような手法は AI ワーカからより多くのタスククラスタを得るために応用できると考えられる．

2.5 フィードバックを伴うタスク設計

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 [39] や Microtalk [40] 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 [41]．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 [42]．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．

2.6　マイクロタスクを通したワーカの能力向上

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 [43]．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 [44]．

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 [35]．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 [45]．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．

本研究の実験では，自己補正にて信頼性の高いワーカから得られた参考回答を提示する場合に，作業に取り組むワーカから得られたタスク結果の品質が改善されるかを明らかにする．ただし，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 [46] [47] [48]．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，ワーカが作業を介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 [49]．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 [50] [51]．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．

第3章　人間＋AIクラウドにおけるタスク結果品質を考慮した動的なタスク割り当て

この章では，人間とAIのワーカで構成される群衆にタスクを動的に割り当てる問題を扱う．AIソフトウェアをタスクに適用するにあたり，AIの性能が認められれば全ての処理を割り当て，そうでなければAIの利用を諦めるという“all-or-nothing”戦略を取るのが一般的である．しかし，このアプローチではAIの性能が十分となるまでAIを活用することができず，また，複数のAIの候補が与えられてもどのように選ぶべきかは自明でない．このように，外部協力者から得られたAIを他のAIや人間ワーカと統合して，より効率的なHuman-AIチームを構築することは困難な課題である．本研究では，AIワーカの評価に“divide-and-conquer”戦略を採用することで，これらの問題に対処する方法を提案する．ここでタスク割り当ては，全体的なタスク結果品質が要求精度を満たしている限り，人間ワーカへのタスク割り当て数が最小であるときに最適である．この問題に対するタスク割り当てアルゴリズムを提案し，幾つかの理論解析およびオープンベンチマークおよび実世界のデータセットを用いた実験について説明する．実験結果は，AIワーカがタスク集合全体を要求精度を満たすように処理できない場合，提案アルゴリズムがベースラインよりも遥かに多くのタスクをAIワーカに割り当てることを示す．さらに，要求精度に応じて，複数のAIワーカに割り当てるタスク数を柔軟に変更できることが明らかとなった．

3.1　はじめに

画像などを対象とした分類タスクの集合$T = \{t_1, \ldots t_M\}$と依頼者により設定された要求精度qが与えられた時に，人間のワーカとAIのワーカに対する最適なタスク割り当てを求める問題を扱う．ここで，タスク割り当ては依頼者の要求精度qを満たした上で，人間ワーカに割り当てられるタスク数が最小であるとき（言い換えればAIワーカに割り当てられるタスク数が最大化される時）最適であると呼ぶ．タスク割り当ては徐々に決定され，人間ワーカによってラベル付けされたタスクは，AIワーカの訓練および評価のためのデータセットとして使われる．したがって，最適なタスク割り当てを求める上で重要となるのが割り当てるタスクを選ぶ順序である．能動学習の研究が示すように，タスクの順序はそれを学習データとする機械学習モ

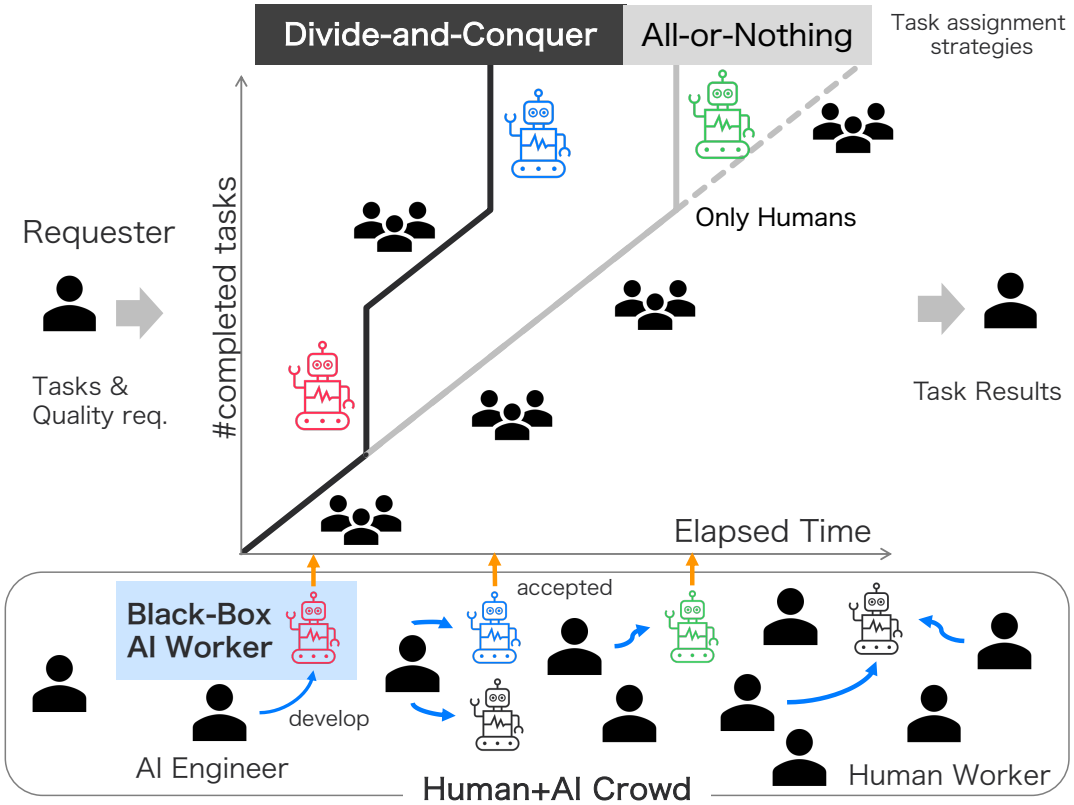


図 3.1: 本研究では，分割統治戦略に基づいて，タスク結果の品質要件を考慮しながら，人間ワーカと AI ワーカにタスクを割り当てる．これにより，能動学習に基づく割り当てなどの “all-or-nothing” 戦略よりも多くのタスクを AI ワーカに割り当てる．(研究業績 2-1 より転載)

デルの性能を大きく左右する．また，タスクの順序は人間ワーカのリソースが限られている状況や急を要する自然災害の状況など，訓練データが不足するドメインで AI を活用するためにも重要である．

タスクに優先順位をつけるための一般的なアプローチは，能動学習を適用して，人間が作業するタスクを決定することである [52]．この方法で AI の品質を保障するためには，テストデータを作成を作成する必要があり，そのためにランダムに選んだタスクを人間に割り当てる必要がある．能動学習で訓練している AI のテストセットに対する出力が要求精度 q を上回る場合，残りの全てのタスクをその AI に割り当てることができる (図 3.1 (上))．ただしこのアプローチには，次の欠点がある．(1) この手法は “all-or-nothing” 戦略によってタスクを割り当てる．そのため，AI ワーカが受け入れられる前に，その AI を活用することはできない．(2) この方法は単一かつ特定の機械学習モデルが事前に入手できていることが前提である．そのため，不特定多数の AI リソースを活用することは難しく，そのプロセスはスケーラブルではない．

そこで，“divide-and-conquer” 戦略によりこれらの問題に対処する方法を提案す

る．提案手法の中心的なアイデアは，AI ワーカが同じクラスに分類するようなタスクの部分集合の単位で AI ワーカの品質を評価することである．これを**タスククラスタ**と呼ぶ．提案手法は AI ワーカのあるタスククラスタが十分な品質を満たしている場合，タスククラスタ内の未ラベルのタスクを AI ワーカ[1]に割り当てる（図 3.1 (下)）．この戦略により，AI ワーカにとって簡単であるタスクの割り当てを早期に決定することができる．つまりタスク集合全体を完了する前に AI ワーカのタスク結果を活用することができる．これは欠点 1 に対処するものである．さらに，提案アルゴリズムは能動学習のアプローチとは異なり，単一の機械学習モデルには依存せず，不特定多数のブラックボックス AI を利用可能である．ブラックボックス AI ワーカは，割り当ての過程でいつでも増減することができる．提案手法では，AI ワーカの出力をタスククラスタとして個別に評価する．したがって，複数の AI ワーカから得られたタスククラスタはタスククラスタの集合として処理されるため，欠点（2）に対処することができる．

本研究の貢献は次の通りである：

(1) 人間＋ AI クラウドタスク割り当て問題: 与えられたタスク集合を人間のワーカと AI のワーカに割り当てる人間＋ AI クラウドタスク割り当て問題を定義する．この問題は，AI ワーカがどのタスクを処理するかを決定するという点で，能動学習とは異なる．さらに，全体的なタスク結果の品質が依頼者の要求精度を満たすことを保証する必要がある．

(2) タスク結果品質に関する理論保証付きのタスク割り当てアルゴリズム: タスク結果の正誤の確率分布がベータ分布で近似できると仮定して，幾つかの理論保障を備えた HACTAP を解決するタスク割り当てアルゴリズムを提案する．

(3) ベンチマークおよび実世界のデータセットを用いた実験: ベンチマークおよび実世界のデータセットを用いて，広範囲な実験を行った．実験結果から，AI ワーカの出力全体が要求精度を満たすことが困難な場合に，アルゴリズムがベースライン手法よりも多くのタスクを AI ワーカに割り当てられることを示した．提案アルゴリズムは，利用可能な AI ワーカの性能に応じて，複数のブラックボックス AI ワーカに割り当てるタスク数を柔軟に変更する．

3.2 人間＋ AI クラウドタスク割り当て問題

この節では，まず問題設定である人間＋ AI クラウドタスク割り当て問題（Human+AI Crowd Task Assignment Problem, HACTAP）を定義する．次に，性能が変動する多数の AI ワーカを扱うように設計された 2 つのタスク割り当てアルゴリズムを提案する．提案アルゴリズムについて，アルゴリズムの出力であるタスク割

[1]本研究での AI ワーカには，ルールベースのアルゴリズムや機械学習モデルなど，あらゆるアルゴリズムに基づくソフトウェアエージェントが含まれる．

表 3.1: Notation

Section	Notation	Description
3.1	$T = \{t_1, \cdots, t_M\}$	A set of given multiple classification tasks with N classes.
3.1	$A = \{a_1, \ldots, a_N\}$	A set of the classes (labels)
3.1	$W = \{w_1, w_2, \ldots\}$, where $w_i : T \to \mathbb{N}$	A set of AI workers implementing any algorithms such as unsupervised, supervised, and rule-based algorithms. Here, we denote each AI worker w_i as a function that returns a natural number representing a cluster assigned by the AI worker to a given task.
3.2	$R_{w_i} = \{(t_j, t_k) \mid w_i(t_j) = w_i(t_k)\}$	The equivalence relation in the set of tasks T by the AI worker w_i where $w_i(t)$ is the label given to t by w_i.
3.2	$C_{w_i} = T/R_{w_i}$	A set of task clusters w_i generates. We define C_{w_i} as the set of sets of tasks with the same predicted label derived by the equivalence relation R_{w_i}.
3.2	$C = \bigcup_{w_i \in W} C_{w_i}$	All task clusters from W at the current assignment progress.
3.3	$ans : T \to \{A \times (W \cup \{\text{`}h\text{'}\})\} \cup \{(\emptyset, \emptyset)\}$	An updatable function that takes a task and returns a pair of the result and the assigned worker. Here, a human worker is denoted by h.

り当てから得られるタスク結果の品質が少なくとも要求精度を上回ることに関して理論解析を行う．

3.2.1 問題定義

表 3.1 にこの章で用いる表記を示す．多値分類タスクの集合を T とする．T の各要素に付与されるラベル候補の集合（各要素は具体的にはクラス名やラベル名）を A とする．匿名の開発者によって作成された実装や性能が未知である AI ワーカの集合を W とする．AI ワーカ集合の各要素 $w \in W$ は関数 $w : T \to \mathbb{N}$ であり，入力のタスクに対して A に限定されないラベルを返す．本研究では，各 AI ワーカの学習および推論の機能のみを任意のタイミングで呼び出すことができるものとする．人間ワーカについては，既存手法を組み合わせることで信頼できる回答が得られることを仮定するため，単に ‘h’ と表記する．

タスクとワーカのペア (t_j, w_i) を要素とするタスク割り当ての列を S とする．S はタスク集合のすべての要素に対するタスク割り当てを含む必要があるため，$|S| = |T|$ である．タスク t_j を担当するワーカ w_i は，AI ワーカの集合の要素または人間ワーカの要素 $w_i \in W \cup \{'h'\}$ である．依頼者によって与えられるタスク結果品質の要求精度を q とする．ただし，$0 \leq q \leq 1$ である．人間+AI クラウドタスク割当問題で求めるのは，全体のタスク結果品質が少なくとも q となるタスク割り当ての集合 S

である．

Definition 1 (人間＋ AI クラウドタスク割り当て問題)．分類タスクの集合 T，AI ワーカの集合 W ，および要求精度 q が与えられ，精度はタスク結果の数のうち人間ワーカのタスク結果と等しいタスク結果の数の割合で定義されるとする．人間＋ AI クラウドタスク割り当て問題とは，タスク結果の精度が要求精度 q を満たすようなタスク割り当て S を求めることである．

HACTAP の解であるタスク割り当て集合の候補を非決定的に作成した後に，その候補が HACTAP の制約を満たせば Yes，そうでなければ No を返すアルゴリズムを考える．ここでは解の候補を求めるためのプログラムの計算量は $\mathcal{O}(1)$ であるとする．解が HACTAP の制約を満たすかどうかは，各タスク $t \in T$ について $ans(t)$ 関数の返り値を調べることで判断できる．つまりこの処理の計算量は $\mathcal{O}(|T|)$ である．よって，解の検証は多項式時間程度で可能である．

HACTAP は言い換えれば，人間ワーカのみに割り当てを行うのと同じタスク結果品質を保ちながら，人間ワーカと AI ワーカへ割り当てを行う別の割り当てを見つけることを目的としていると言える．つまり，この問題には全てのタスクを人間ワーカに割り当てるという単純な解が存在することに注意しなければならない．

ここで，タスク割り当て S_1 が 別のタスク割り当て S_2 よりも効率的であることを説明する．S_1 は $(t_j, `h')$ であるような要素の数が S_2 よりも少なく，かつ全体的なタスク結果品質が q を満たしているとする．ここで，q を満たすタスク割り当ての中で S_1 よりも効率的なタスク割り当てが存在しない場合，S_1 は最適であると言える．

一方で最適解を求めるためには，人間ワーカに全てのタスクを割り当てることを前提として，人間ワーカへのタスクの割り当て順の全ての組み合わせにおいて，AI ワーカへのタスク割り当ての全ての組み合わせを検討する必要がある．人間ワーカのタスク結果を AI ワーカの学習およびテストデータとしてどのように分割するかも，割り当て結果を左右する要因として挙げられる．列挙したタスク割り当ての候補において，AI ワーカへのタスク割り当て数が最大となるような候補が最適解である．このアルゴリズムの計算量は $\mathcal{O}(|T|!|C|!)$ となる．この戦略では全てのタスクを人間ワーカに割り当てる必要があり，実用的ではない．これらの理由から，全ての組み合わせを検討することなく，割り当てを動的に決定するアルゴリズムが必要である．

提案手法では，人間ワーカへの割り当てと，AI ワーカの評価と割り当てを交互に行うことで，少なくとも HACTAP の制約を満たすような割り当てを求める．提案手法に基づくアルゴリズムが最適解を出力するとは限らない．提案手法は少なくとも要求精度を満たす解を得ることを保証するアルゴリズムであるが，近似性能を保

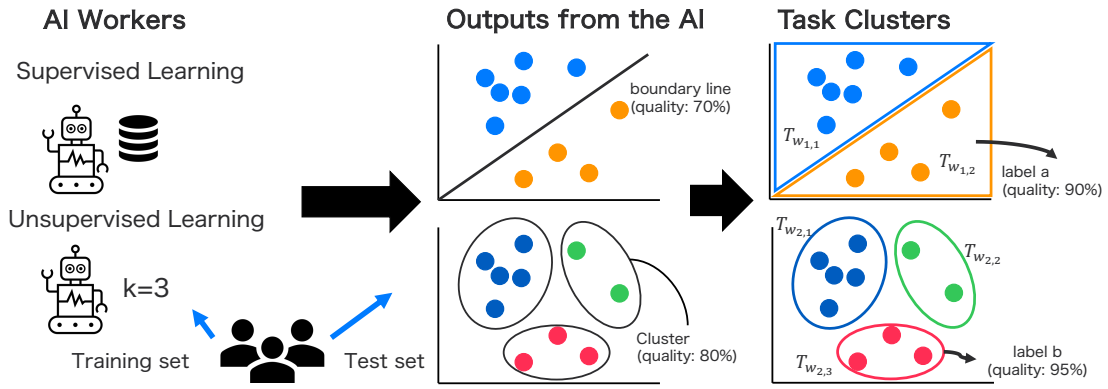


図 3.2: 多様な AI ワーカからの出力を，タスククラスタ（タスクの部分集合）の単位で評価することで，各 AI ワーカが得意とする処理を発見する．（研究業績 2-1 を改変）

証するものではない．この意味で，提案アルゴリズムは発見的解法であるといえる．提案アルゴリズムの計算量は $\mathcal{O}(|T||C|)$ である．

3.2.2 タスククラスタ

提案アルゴリズムを説明する前に，HACTAP で AI ワーカに割り当てるタスク数を最大化するためのアイデアであるタスククラスタを導入する．タスク全体に対して十分な精度を持つ AI ワーカが存在する場合，その時点での残タスクを全てその AI ワーカに割り当てるのが有効である．しかしながら，そのような理想的な AI ワーカが常に存在するとは限らず，構築までに時間やより多くの訓練データを要する可能性が高い．そこで，類似タスクからなるタスククラスタ（タスク集合の部分集合）を構成し，AI ワーカをタスククラスタ毎に評価する．実験結果が示すように，タスククラスタのアイデアに基づくアルゴリズムは，all-or-nothing 戦略よりも多くのタスクを AI ワーカに割り当てることができる．

各 AI ワーカの出力をタスククラスタの集合として定式化するための全体像を図 3.2 に示す．各タスククラスタは AI ワーカによって同じ種類のラベルが付与されたタスク集合の部分集合である．各 AI ワーカは訓練データとしてタスクとラベルのペアの集合を入力とし，テストセットのタスクを k 個のタスククラスタに分割する．ここで，k はラベル集合の要素数 $|A|$ に依存するとは限らない．これは，例えば教師なしクラスタリングアルゴリズムが AI ワーカとして参加する可能性があり，このような AI ワーカは A に対応しない出力を返すからである．

AI ワーカとは対照的に，人間ワーカにおいてタスククラスタを扱うのは現実的ではない．通常の機械学習の文脈とは対照的に，AI ワーカはタスク全体に高い精度を持つ必要はない．各タスククラスタを評価し，要求精度に対して品質が十分であれば，タスククラスタに含まれるようなタスク結果をその AI ワーカから受け入れるこ

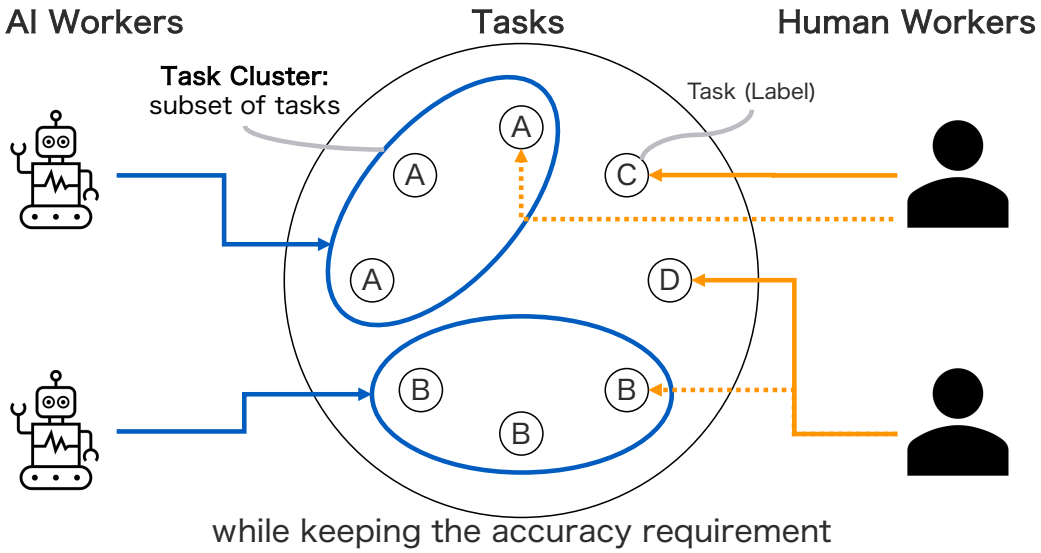


図 3.3: 提案手法は全体的なタスク結果品質を考慮しながら，人間ワーカと AI ワーカにタスクを割り当てる．(研究業績 2-1 を改変)

とができる．この戦略によって，要求精度と AI ワーカに割り当てるタスク数においてきめ細かいトレードオフ制御を実現できる（図 3.3）．

表 3.1 にタスククラスタの表記の概要を示す．AI ワーカ w_i によるタスク集合 T の同値関係を $R_{w_i} = \{(t_j, t_k) \mid w_i(t_j) = w_i(t_k), t_i, t_j \in T, t_i \neq t_j\}$ とする．AI ワーカ w_i から得られるタスククラスタの集合は T の商集合 $C_{w_i} = T/R_{w_i} = \{T_{w,0}, T_{w,1}, ..., T_{w,\mathbb{N}}\}$ として定義される．前述のとおり，タスクラスタの数 $|C_{w_i}|$ は AI ワーカによって異なる．多くの教師あり学習アルゴリズムでは $|C_{w_i}|$ は $|A|$ となるが，それ以外の場合も考えられる．例えばクラスタリングアルゴリズムが T を k クラスタに分割する場合，$|C_{w_i}|$ は k となる．AI ワーカ集合 W から得られる全てのタスククラスタは $C = \bigcup_{w \in W} C_w$ と表記する．

タスククラスタ単位での精度の評価を行うことにより，AI ワーカからの出力を部分的に採用するか否かの判断が可能になる．これにより，特性が多種多様であり，全体的な性能が十分とは限らない複数の AI ワーカにタスクを割り当てることが期待できる．

3.2.3 Clusterwise Test-based Assignment (CTA)

CTA は，人間ワーカからのタスク結果を利用して AI ワーカから得られた各タスククラスタを統計的に評価し，その結果に基づいて各ワーカへの割り当てを決定するアルゴリズムである．CTA では統計的検定により，対象のタスククラスタに含まれる人間ワーカラベルについて最頻出ラベルの出現割合が q を上回るか評価する．統計的検定によりタスククラスタの品質が認められた場合，タスククラスタ中の未

Algorithm 1 Clusterwise Test-based Assignment (CTA)

Require: A set T of tasks, a set W of AI workers, the accuracy requirement q, and the significance level α.

Ensure: A sequence of (task, worker) pairs.

1: **for all** $t \in T$ s.t. $ans(t) = (\emptyset, \emptyset)$ in a random order **do**
2: $a' \leftarrow task\_result(assign(t, \text{`}h\text{'}))$
3: update ans so that $ans(t) = (a', \text{`}h\text{'})$
4: **for all** $T_{w_i,j} \in C$ **do**
5: $\hat{a} = \underset{a \in A}{arg\,max} \, |\{t \mid t \in T_{w_i,j}, ans(t) = (a, \text{`}h\text{'})\}|$
6: **if** $statistical\_test(T_{w_i,j}, \hat{a}, q, \alpha)$ **then**
7: for $t' \in T_{w_i,j}$ s.t. $ans(t') = (\emptyset, \emptyset)$, $assign(t', w_i)$ and update ans so that $ans(t') = (\hat{a}, w_i)$
8: **end if**
9: **end for**
10: **end for**

ラベルのタスクに対して最頻出ラベルを付与する．この処理はAIワーカからのタスク結果を受け入れることに相当する．

入出力　CTAは入力として，タスク集合 T，AIワーカ集合 W，要求精度 q および有意水準 α を受け取る．CTAが出力するのは，タスクとワーカのペアで構成されるタスク割り当ての列 $[(t_1, w_{t_1}), \ldots, (t_k, w_{t_k})]$ である．このタスク割り当ては，アルゴリズム中の $assign(t, w)$ 関数の実行履歴に相当する．

手続き　アルゴリズム 1 にCTAの手続きを示す．CTAは，$assign(t, w)$ を実行しながら，割り当てによって得られたタスク結果 $a \in A$ を用いて ans 関数を更新する．$ans : T \to \{A \times (W \cup \{\text{`}h\text{'}\})\} \cup \{(\emptyset, \emptyset)\}$ は更新可能な関数であり，入力としてタスクを受け取り，タスク結果とそのタスクが割り当てられたワーカを返す．

初期状態では，$ans(t)$ は全ての $t \in T$ に対して $(\emptyset, \emptyset)$ を返す．これはどのタスクにも割り当てが行われておらず，タスク結果がないことを意味する．

CTAは，$ans(t)$ が全ての t に対して $(\emptyset, \emptyset)$ を返さなくなるまで ans 関数の更新を続ける．ここで，$ans : T \to A \times (W \cup \{\text{`}h\text{'}\}) \cup \{(\emptyset, \emptyset)\}$ はタスクを入力するとラベルとワーカのペアを返す更新可能な関数である．あるタスクに対するラベルとワーカが未定の場合は $(\emptyset, \emptyset)$ を返す．CTAはまず，割り当てが決定していないようなタスク，つまり $ans(t) = (\emptyset, \emptyset)$ であるような t をランダムに選択し，$assign$ 関数によって人間ワーカに割り当てられ，$task\_result$ 関数によってタスク結果を取得する（2行目）．人間ワーカからタスク結果（例えば `a` や `b` などの具体的なラベル）を受け取る（3行目）と，CTAは次の一連の手続きを実行する（4行目から9行目．図3.2に相当）．各タスククラスタ $T_{w_i,j} \in C$ に対して，統計的検定を行う．ここで，$T_{w_i,j}$ はAIワーカ w_i によって作成された j 番目のタスククラスタを意味する．5

行目では，タスククラスタ中の人間ワーカによるラベル済みタスクのうち，最頻出ラベルを $\hat{a}$ とおく．6 行目では $statistical\_test$ 関数により，タスククラスタの各タスクのラベルを $\hat{a}$ とした場合に，タスククラスタから得られるタスク結果品質が要求精度 q を有意水準 α で上回るかを判定する．この統計的検定では，すでに人間ワーカによって与えられているタスク結果を用いて，$T_{w_i,j}$ の各タスクのラベルが $\hat{a}$ である比率が要求精度 q より大きいかを確かめる．ここで $\hat{a}$ は，そのタスククラスタに含まれる人間ワーカに割り当て済みのタスクにおいて出現頻度が最も高いラベルである．ここでは，$\hat{a}$ の割合が有意水準 α において q よりも高いことを統計的に検定できるような任意の統計的検定を適用することができる．本論文の実験では，$statistical\_test$ 関数として二項検定を用い，タスククラスタから得られるタスク結果品質と要求精度の間に差がないことを帰無仮説として，片側検定を行った．試行回数 $n_{w_i,j}$ と成功回数 $n^{(p)}_{w_i,j}$ はそれぞれ次のように求めた．帰無仮説が棄却された場合，タスククラスタ中の未ラベルであるタスクの ans 関数を $(\hat{a}, w)$ で更新する（7 行目）．

$$
\begin{aligned}
n_{w_i,j} &= |\{t \in T_{w_i,j} \mid ans(t) = (a, \text{`}h\text{'}), a \in A\}| \\
n^{(p)}_{w_i,j} &= |\{t \in T_{w_i,j} \mid ans(t) = (\hat{a}, \text{`}h\text{'})\}|
\end{aligned}
$$

検定により $\hat{a}$ の出現確率が q よりも高いと認められた場合，タスククラスタ中の各タスク $t \in \{t' \in T_{w_i,j} \mid ans(t') = (\emptyset, \emptyset)\}$ について $ans(t)$ を $(\hat{a}, w_i)$ で更新する．

ここで，CTA が求めるタスク割り当てについて理論解析を行う．

Theorem 1 (CTA による割り当てで得られるタスク結果品質)．*CTA* は，$(1-\alpha)^l$ の確率で全体的なタスク結果品質が要求精度 q を満たすようなタスク割り当てを計算する．ただし，l は統計的検定の回数を，α は各統計的検定で用いる有意水準とする．

証明．CTA が要求精度を満たすタスク割り当てを出力する確率は，family-wise error rate (FWER) に基づいて計算できる． □

定理 1 は，特にタスククラスタ数が多い状況下で，要求精度を満たすのが困難であることを示している．一般的に，統計的検定を複数回実施する場合には，ボンフェローニ補正などの方法で各統計的検定での有意水準を補正することで FWER を制御する．しかし HACTAP ではタスク割り当てが完了するまで，タスククラスタの検定回数が不明なので，これらの方法で FWER を制御することが困難である．この問題に対処する方法として（１）事前に統計的検定の回数を見積り，見積もりに応じて FWER が許容できるように実際の有意水準を調整する（２）オンライン FWER 制御の手法 [53] を適用することが挙げられる．

Algorithm 2 The *statistical_testing* function for GTA

Require: A task cluster candidate $T_{w_i,j}$, the accuracy requirement q, and the significance level α.
Ensure: True if $T_{w_i,j}$ is acceptable and false otherwise.

let $\Gamma_{accepted}$ be a set of accepted task clusters
let $\Gamma = \Gamma_{accepted} \cup \{T_{w_i,j}\}$
let n be the number of iterations
let $v_{T_i,j} \sim Beta(1 + T_i.r, 1 + T_i.c) \forall T_i \in \Gamma, 1 \leq j \leq n$
$success = 0$
for j in range(n) **do**
    $acc_j = \frac{\sum_{T_i \in \Gamma} v_{T_i,j} T_i.size}{\sum_{T_i \in \Gamma} T_i.size}$
    if $acc_j \geq q$ **then**
        $success = success + 1$
    end if
end for
return $(1 - (success/n)) < \alpha$

タスク数が大きい場合，AI ワーカから受け入れるタスククラスタの数，つまり検定の回数 l が多くなる傾向があり，これに伴い統計的検定のエラー率が増加する．この問題は，各タスククラスタが個別に評価する仕組みによる物である．次に説明する，全体的なタスク結果品質を考慮した割り当てではこの問題を解決するためのタスククラスタの評価方法を提案する．

3.2.4 Global Test-based Assignment (GTA)

ここでは，タスク結果の全体的な精度を理論保証するタスク割り当てアルゴリズムについて説明する．精度が少なくとも q であるかどうかを検証するために各タスククラスタをテストする CTA とは対照的に，GTA は全てのタスク結果の全体的な精度を考慮して割り当てを行う．GTA では，各タスククラスタから得られるタスク結果の品質に基づいて，全体的なタスク結果品質の確率分布を直接計算する．

全体的なタスク結果品質の確率分布の計算方法を説明する．GTA では，各タスククラスタから得られるタスク結果品質の確率分布はベータ分布としてモデル化されている．ある時点までに得られた人間ワーカからのタスク結果に基づいて，beta-binomial conjugacy を適用することで，各タスククラスタの品質の事後分布を計算することができる．タスククラスタの部分集合を $\Gamma = \{T_1, T_2, \ldots\}$ とする[2]．タスククラスタ T_i の精度を示す確率変数を V_{T_i} とする．V_{T_i} の事前分布は一様な分布に従うと想定され，この時 $P(V_{T_i}) = Beta(1, 1)$ である．V_i の事後分布は，$P(V_{T_i} \mid T_i) = Beta(1 + T_i.r, 1 + T_i.c)$ と計算することができる．ただし，$T_i.r$ は人間ワーカの最頻ラベルと同じラベルであるタスクの数，$T_i.c$ は最頻ラベルとは異なるラベルである

[2] ここではタスククラスタの表記として $T_{w_i,j}$ の代わりに T_k を用いる．タスククラスタの評価においてはどの AI ワーカから得られたタスククラスタかを区別する必要がないからである

タスクの数である．$\bigcup_{T_i \in \Gamma} T_i$ の全体的な精度を Acc とする．これは，V から変換された確率変数であり，次のように定義される．

$$Acc = \frac{\sum_{T_i \in \Gamma} V_{T_i} T_i.size}{\sum_{T_i \in \Gamma} T_i.size}, \tag{3.1}$$

ただし $T_i.size$ は $|\{t' | t' \in T_i, ans(t') = (\emptyset, \emptyset)\}|$. 次に，全体的な精度 $P(Acc \mid \Gamma)$ の分布を評価することで，タスク割り当てにより得られる精度が要求精度を満たしている確率を計算できる．

$$P(Acc < q \mid \Gamma) = \int_0^q P(Acc = a \mid \Gamma) da. \tag{3.2}$$

全体的な精度が有意水準 α で q を満たしているならば，$P(Acc < q \mid \Gamma) < \alpha$ が成立しなければならない．

Global Test-based Assignment (GTA)　ここで，CTA のアルゴリズム（アルゴリズム 1）を拡張して GTA を定義する．GTA では，アルゴリズム 1 の 5 行目で呼び出す *statistical_test* においてタスククラスタ毎の検定を実行する代わりに，全体的な評価に基づく検定を行う．アルゴリズム 2 に，GTA で用いる *statistical_test* の手続きとして用いるモンテカルロシミュレーションのアルゴリズムを示す．GTA は実行の過程において $\Gamma_{accepted}$ を保持する．これは人間ワーカのタスク結果を含む特別なタスククラスタ T_{human} を含む，すでに採用済みのタスククラスタの集合である．

ここで，GTA で得られるタスク結果品質について理論解析を行う．

Theorem 2 (Correctness of GTA). シミュレーションの試行回数である J が無限に近付くにつれ，*GTA* は全体的なタスク結果品質が q を上回るようなタスク割り当てを出力する．

証明. 確率密度関数 $P(acc)$ の期待値を解析的に求めることはできないが，V_i の事後分布分布 $Beta(1 + T_i.r, 1 + T_i.c)$ からサンプルを取得できるので，モンテカルロ積分により期待値を求めることができる．モンテカルロ法を用いて $P(acc > q \mid \Gamma)$ の期待値を次のように概算できる：

$$\frac{1}{J} \sum_{j=1}^{J} \delta \left(\frac{\sum_i R_{i,j} T_i.size}{\sum_i T_i.size} < q \right), \tag{3.3}$$

ただし，$\delta(\cdot)$ は $\cdot$ が真なら 1 を，それ以外は 0 を返すデルタ関数である．式 3.1 と式 3.2 の仮定に基けば，この値は $J \to \infty$ のとき $P(acc < q \mid \Gamma)$ に収束する．ただし J はモンテカルロシミュレーションの試行回数とする．このシミュレーションはアルゴリズム 2 に相当するため，GTA はこれらの仮定のもので，全体的な要求精度が q を上回ることを保証する．　□

3.3 ベンチマークデータセットを用いた実験

ここでは，ベンチマークデータセットを用いた実験により，(1) 提案アルゴリズムにより得られるタスク結果品質が要求精度を満たしているか (2) 提案アルゴリズムが要求精度に応じて，タスク結果品質とコスト（人間ワーカに割り当てられるタスクの数）の間のトレードオフをどのように調節するかを明らかにする．

3.3.1 実験設定

くずし字データセット [54] から 10,000 枚の画像をランダムに選択し，10 クラス分類タスク（$|A| = 10$）を作成した．実験で使う AI ワーカは，機械学習ライブラリである scikit-learn 0.23.1 にて実装されている 15 種類の機械学習モデルを初期設定で利用した．

実験では，２種類のベースラインアルゴリズム（Worker-wise Random Sampling Test based assignment (WTA)，Active learning-based assignment (ALA)）と提案手法により比較実験を行う．これらのベースライン手法は，AI ワーカの全体的な手法を評価する．

複数の AI ワーカを扱うために，weighted voting ensemble 法により各 AI ワーカの出力を統合する．さらに ALA では，統合して得られたタスク結果に基づいて次のタスクを決定する．そのため，各 AI ワーカは単にラベルだけでなく，予測確率を返す必要がある．実験では 15 種類の AI ワーカのうち 10 種類の AI ワーカのみが利用可能である．ALA は能動学習に基づくアルゴリズムであることから，並列処理が困難である，クラウドソーシングプラットフォームとの同的な相互作用が求められる．

付録に実験で使用する AI ワーカの一覧を示す．

3.3.2 アルゴリズムの詳細

実験で比較するアルゴリズムの詳細を説明する．

Worker-wise Random Sampling Test-based Assignment (WTA)　WTA は，AI ワーカ集合を統合して得られた出力が要求精度を上回るかを評価し，その結果に基づいて AI ワーカにタスクを割り当てるかどうか決定するアルゴリズムである（アルゴリズム 3）．実験では複数の AI ワーカが利用可能な場合，重みつき多数決により AI ワーカ集合の出力を統合した．

Active Learning-based Assignment (ALA)　ALA は，能動学習に基づいたタスク割り当てアルゴリズムである．ALA において人間ワーカに割り当てるタスクのうち，

Algorithm 3 Worker-wise Random Sampling Test based assignment (WTA)

Require: A set T of tasks, an AI worker w, and the accuracy requirement q.

Ensure: A sequence of (task, worker) pairs.

1: **while** $accuracy(w) < q$ or $\exists t \in T$ s.t. $ans(t)! = (\emptyset, \emptyset)$ **do**
2: let $t \in T$ s.t. $ans(t) = (\emptyset, \emptyset)$ in a random order
3: $a' \leftarrow task\_result(assign(t, `h'))$
4: update ans so that $ans(t') = (a', `h')$
5: **end while**
6: for all $t' \in T$ s.t. $ans(t') = (\emptyset, \emptyset)$, $assign(t', w)$ and update ans so that $ans(t') = (w(t'), w)$

半数はクエリ戦略により決定されたものであり，残りはランダムに選択したものである．ALA はWTA のアルゴリズムの２行目での人間ワーカへの割り当てを変更することで実現できる．

アルゴリズム中では，クエリ戦略により人間ワーカにタスクを割り当てて得られたタスクとラベルの集合を用いて，AI ワーカを訓練し，ランダムな割り当てで得られたラベルを用いて AI ワーカを評価する．AI ワーカの出力の品質が要求精度を上回った場合，その時点での残りのタスクを全て AI ワーカに割り当てる [52]．ALA は提案アルゴリズムとは異なり，予測確率を出力できる必要があるため任意のブラックボックス AI ワーカを利用することは出来ず，AI ワーカはアルゴリズムの開始時に参加している必要がある．

Clusterwise Test-based Assignment (CTA)　実験では，アルゴリズム 1 の６行目において二項検定によりあるタスククラスタ $T_{w_i,j}$ が $\hat{a}$ として採用できるかを判断する．ここで，試行回数 n は $n = |\bigcup_{t \in T_{w_i,j}} ans(t) = (*, `h')|$ により，成功回数 $m$ は $m = |\bigcup_{t \in T_{w_i,j}} ans(t) = (\hat{a}, `h')|$ により求める（ただし，*は任意のラベルを表す）．

Global Test-based Assignment (GTA)　モンテカルロシミュレーションの試行回数は $n = 100,000$ とした．

実験では，人間ワーカに 200 個のタスクを割り当て，その結果が得られた際に AI ワーカの学習および評価を実行した．つまり，異なる t についてアルゴリズム 1 の２行目と３行目を繰り返す．CTA と GTA の有意水準は $\alpha = 0.05$ とした．

アルゴリズムにおいてタスククラスタや AI ワーカの評価順序は割り当て結果を左右する要因である．割り当て順序は AI ワーカに対する報酬設計に拡張できる可能性がある．本研究では，要求精度を満たすタスク割り当てを得ることに焦点を当てているため，ランダムな順序でタスククラスタを評価した．

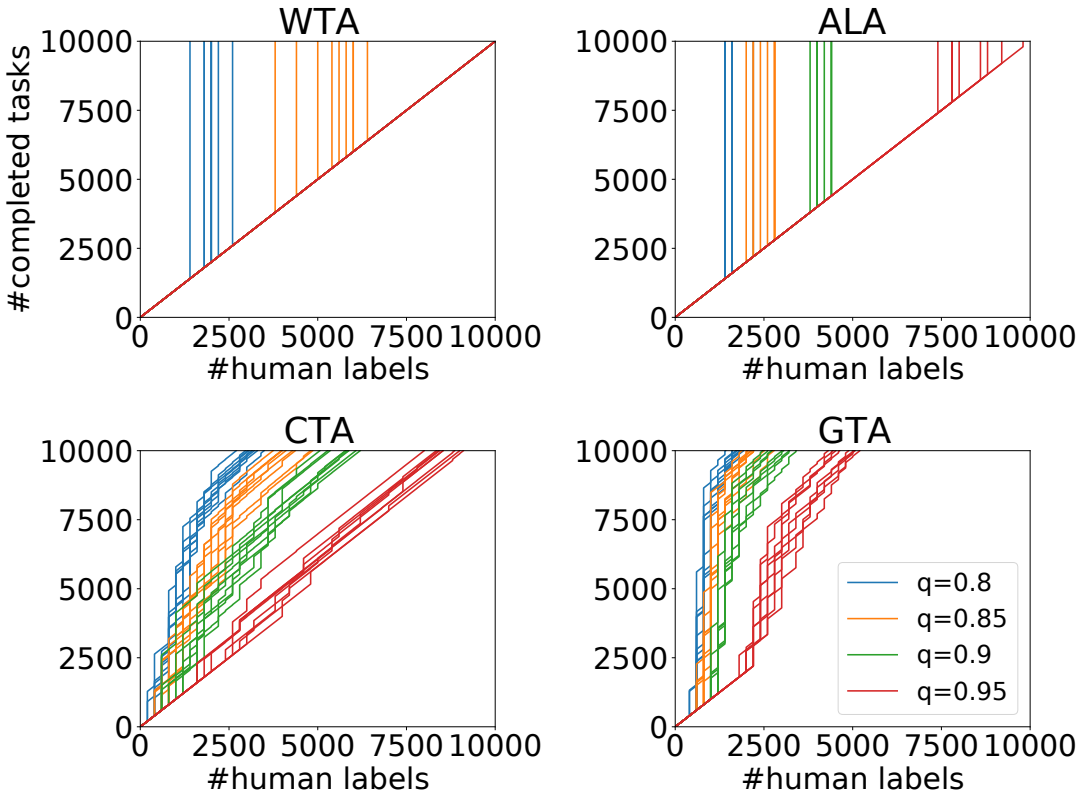


図 3.4: KMNIST を用いた実験結果：各手法における人間ワーカに割り当てられたタスク数と全体の完了タスク数の関係.（研究業績 2-1 より転載）

3.3.3 実験結果

実験では，アルゴリズムの各パラメータの設定で 100 試行を実行した．実験環境は Ryzen 9 3950X 16-Core Processor, 64 GB RAM, GeForce RTX 3090 GPU, Ubuntu 18.04, Python 3.8.2 である．

図 3.4 に実験結果を示す．上段のグラフは，各アルゴリズムの結果における，人間ワーカが完了したタスク数と人間および AI ワーカが完了したタスク数の関係を示す．同じ色の線は，特定の精度要件 q の異なる試行結果を意味する．グラフには 100 試行のうちランダムに選んだ 10 試行のみを含む.

全体として，各アルゴリズムは要求精度を満たすようにタスク割り当てを変化させながら動作した．要求精度が低くなるにつれ，AI ワーカにより多くのタスクが割り当てられた．実験結果から次の 3 つのことが明らかになった.（1）CTA と GTA は，AI ワーカの全体的なタスク結果品質が要求精度を満たすのを待たずにタスククラスタ単位での割り当てを行うため，全体としてより少ない人間ワーカへの割り当て数で完了することができる.（2）ALA は WTA と比較して，より高い要求精度の設定でも AI ワーカにタスクを割り当てることができた．これは，ALA では AI ワー

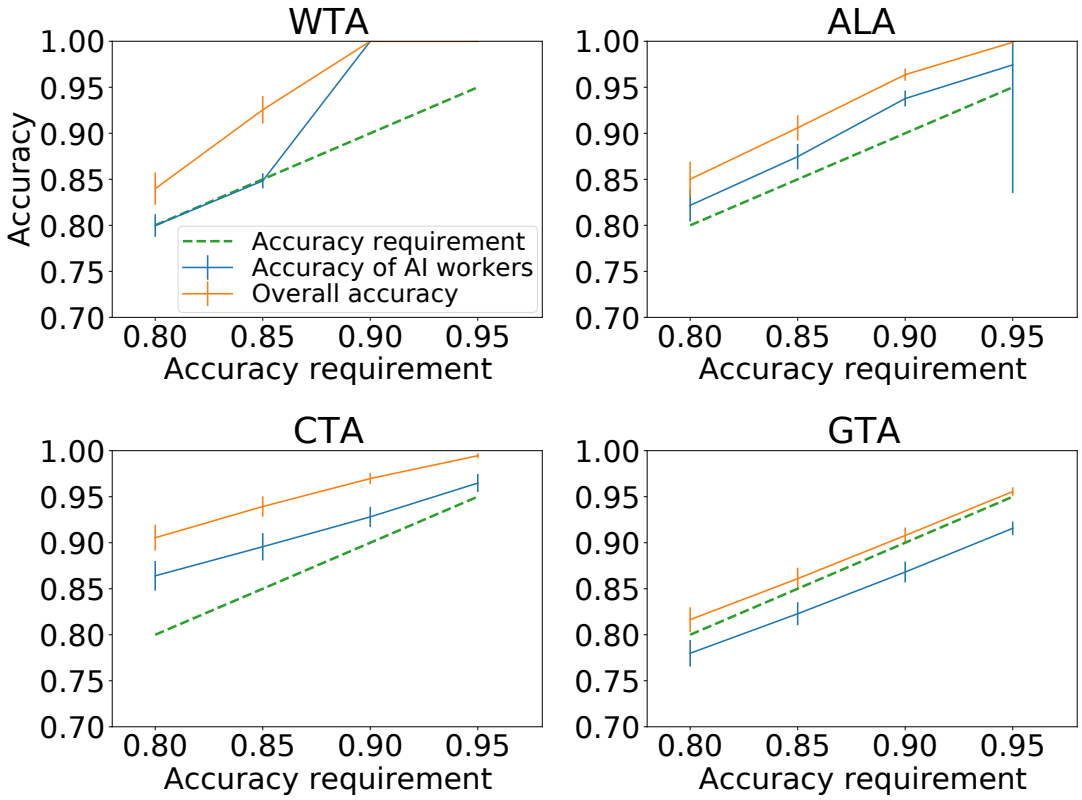


図 3.5: KMNIST を用いた実験結果：各手法における要求精度毎の全体的なタスク結果品質と AI ワーカが処理した部分のタスク結果品質（エラーバーは標準偏差を意味する）.（研究業績 2-1 より転載）

カが効率的に学習できるためである.（3）ALA は CTA よりも AI ワーカにタスクを割り当てる場合がある．特に，要求精度が低い設定（$q = 0.8, 0.85$）でこの傾向が見られた．具体的には要求精度が 0.9 のとき（青い線），CTA が全タスクを完了するまでに 2161-3780 タスクを人間ワーカに割り当てたのに対して，ALA は 1000-2200 タスクを人間ワーカに割り当てた．これは，ALA は AI ワーカが選択した人間ワーカラベルを用いて学習できること，CTA は AI ワーカを統計的に評価するために，人間ワーカのタスク結果を必要とするからだと考えられる．特に，要求精度が高くなく，能動学習により要求精度を容易に達成できる場合，ALA は AI ワーカを早い段階で受け入れることができる．この結果から分かるように，AI ワーカが要求精度よりも高い品質でタスク全体を処理できる場合，all-or-nothing 戦略はうまく機能する.（4）GTA は特に要求精度が高い条件で，4 つのアルゴリズムの中で最も優れた性能を発揮した．一方で，GTA は要求精度が低い場合でも，ALA と同等またはそれ以上の性能を発揮した．次に示すように，GTA は要求精度が満たされている限り，AI ワーカに割り当てるタスク数を最大化する．これはタスククラスタ戦略により全体的なタスク結果品質を細かく制御できるからである.

図 3.5 は，要求精度 q での実験毎の全体的なタスク結果品質を示している．実線はタスク結果全体の品質と AI ワーカが処理した部分のタスク結果品質を示している．点線は，要求精度を表している．

タスククラスタ数が多い状況では保証が難しくなるにもかかわらず，CTA から得られる全体的なタスク結果品質は常に要求精度を上回った．GTA も同様に要求精度を満たすが，全体的なタスク結果品質はより要求精度に近くなる．これは，GTA は全体的なタスク結果品質を考慮しながら，タスククラスタを評価するからである．

実験結果を要約すると，次のことが明らかとなった．（1）WTA と ALA は概ね HACTAP の要求精度を満たすように振る舞うことができたが，要求精度が高い設定では要求精度を満たせない，もしくはより多くの人間ワーカへの割り当てを必要とした．（2）弱い保証のある CTA はタスククラスタから得られるタスク結果品質が要求精度を上回るかを評価するため，要求精度を満たすことはできるが，全体としては要求精度を遥かに超えてしまうことがある．（3）理論保証のある GTA は，他のアルゴリズムよりも多くのタスクを AI ワーカに割り当てながら，適切に動作した．

CTA はタスク割り当ての終盤で人間ワーカにタスクを割り当てる傾向があるが，一方で GTA では AI ワーカに割り当てる傾向が見られた．これは，AI ワーカに実際に求められる要求精度が，人間ワーカへのタスク割り当て数の増加に伴い減少するためである．

3.4 実世界のタスクを用いた実験

ここでは，実世界のタスクで提案アルゴリズムの性能を評価するために，国際防災訓練[3]で使われたタスクを用いた実験を行う．自然災害が発生した状況下では，被災地の状態が急速に変化する可能性があり，素早く意思決定を行う必要がある．そのため，政府や自治体などの依頼者はできるだけ早くタスク結果を入手する必要がある．どのようなシステムが必要かを事前に予測することは難しいが，多くの場合時間を要するため，発災後にシステム構築を開始するのでは間に合わない場合も考えられる．このような状況では，人間ワーカと AI ワーカの自動的な統合が効果的であると考えられる．

3.4.1 設定

実験では，国土地理院が撮影した 2018 年の西日本大洪水の被災箇所の航空写真を使用した[4]．公開されている 106 枚の航空写真のうち，被災箇所を含む 10 枚の画像を用いた．実験では各画像を 1024 個の画像に分割し，各分割画像を 1 つのタスクとし

[3] https://crowd4u.org/events/mind-cndd/index.html
[4] https://www.gsi.go.jp/BOUSAI/H30.taihuu7gou.html

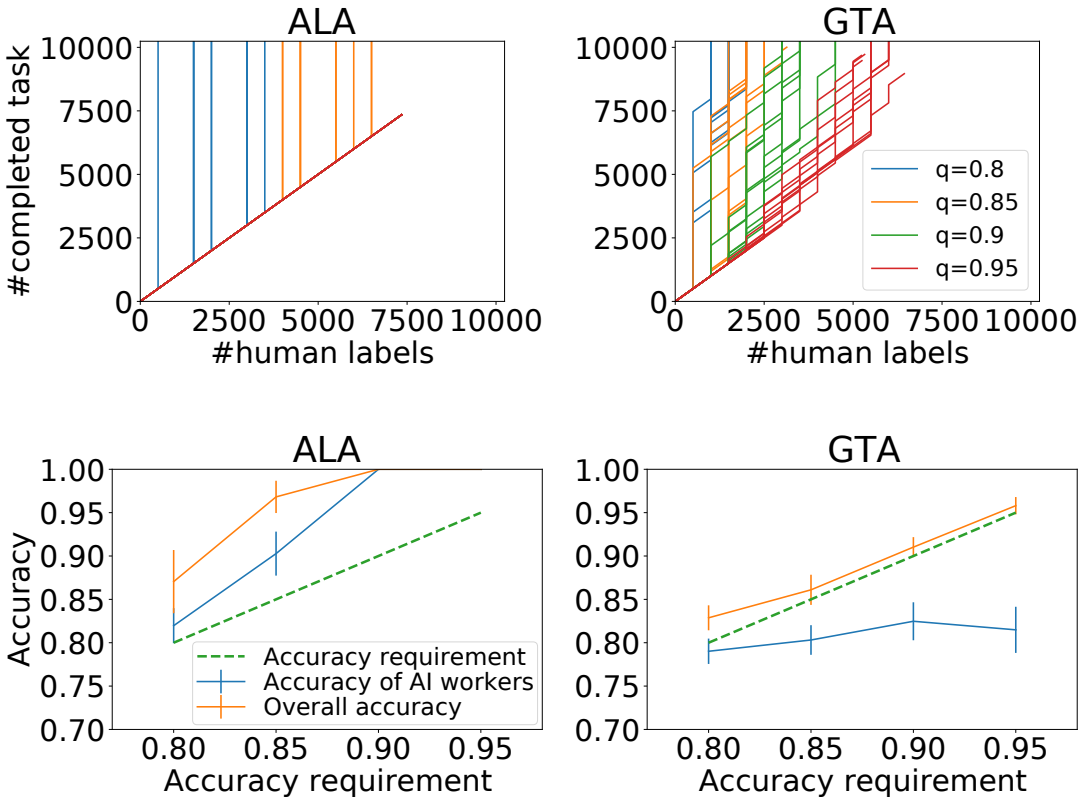


図 3.6: 実世界のデータセット（水害被害判定タスク）での実験結果（研究業績 2-1 より転載）

た．つまり，全体では 10240 個のタスクを実験で用いた．実験では画像を 3 クラスに分類する．対象とするラベルは「浸水していない」「浸水している」「雲で覆われている」である．人間ワーカは Amazon MechanicalTurk により公募した．人間ワーカから得られたタスク結果のクラス毎の内訳は，4736 件の「浸水していない」ラベル，2327 件の「浸水している」ラベル，282 件の「雲に覆われた」ラベルであった．残りの 2895 件にはラベルが付与されていない．

実験では，深層学習に基づく 3 種類の分類モデルが AI ワーカとして参加した．AI ワーカの 1 つは，商用クラウドソーシングサービスを通して，前述の防災訓練のために匿名の開発者によって開発されたものである．本研究では AI ワーカはブラックボックスであることが前提なので，AI ワーカの中身を明らかにする必要がないが，実験のため実行可能な Python および Keras で記述されたコードを受け取った．AI ワーカ開発者には 2 時間の作業に対して 240 米ドルを報酬として支払った．この作業時間は開発者の自己申告によるものである．その AI ワーカは，convolution, pooling, dense, and dropout layers を含む 8 層のネットワークで構成されていた．他の 2 種類の AI ワーカは，Pytorch に含まれている画像分類モデルである ResNet-18 および VGG-16 であった．

3.4.2　結果

3.3 節と同じ実験環境で，要求精度毎に 10 試行行った．

実験結果から，3.3 節と結果と同様の傾向が見られた．GTA は要求精度が高い（$q = 0.9, 0.95$）場合に，ALA よりも多くのタスクを AI ワーカに割り当てた．ただし，要求精度が低い（$q = 0.8, 0.85$）場合は，AI ワーカの出力の全体的な品質が要求精度に到達する可能性があるため，ALA が GTA を上回る場合が見られた．図 3.6 の上段のグラフは，各要求精度 $q \in 0.8, 0.85, 0.9, 0.95$ での ALA と GTA の結果を示している．各グラフは人間ワーカが完了したタスク数と，人間ワーカおよび AI ワーカが完了したタスク数の関係を示している．

下段のグラフは，要求精度毎の実験での実際のタスク結果品質を示している．$q = 0.8$ と 0.85 の設定では，GTA と ALA の両者が，要求精度を満たしながら，AI ワーカに同程度の数のタスクを割り当てた．$q = 0.9$ と 0.95 の設定では，AI ワーカが要求精度を上回らないため，ALA は全てのタスクを人間ワーカに割り当てた．一方で，GTA は AI ワーカからのタスク結果を採用することができた．タスククラスタから得られる実際の品質は要求精度を下回るものであったが，GTA は全体的なタスク結果品質を考慮してそれらのタスククラスタを採用することができた．

図 3.2 は，タスク結果を実際の航空写真上で可視化した結果を表している．(A）はオリジナルの画像を，(B）は人間ワーカから得られた，今回は正解ラベルとみなす結果である．(C）と（D）は，$q = 0.85$ での ALA と GTA が人間ワーカと AI ワーカにどのようにタスクを割り当てたかを表している．GTA は，AI ワーカが森林で覆われているような浸水していない箇所の識別が得意であることを発見し，すぐにそれらのクラスのタスクを AI ワーカに割り当てた．これにより，人間ワーカは，判断がより難しい領域のタスクが割り当てられ，それらの判断に注力できた．

3.5　まとめ

依頼者の要求精度を満たすように，人間ワーカおよび AI ワーカにタスクを割り当てる，人間＋ AI クラウドタスク割り当て問題（HACTAP）を導入した．HACTAP を解決するために，AI ワーカの出力の品質を全体的に評価する代わりに，出力の一部（タスククラスタ）を評価するアイデアに基づくタスク割り当てアルゴリズムを提案した．さらに，提案アルゴリズムに対して幾つかの理論解析を与えた．

実験結果から，アルゴリズムにより得られるタスク結果品質が要求精度を満たし，アルゴリズムが要求精度に応じて AI ワーカに割り当てるタスク数を柔軟に変更できることが示された．

本研究の結果は，AI ワーカと人間ワーカが適切に作業を分担することで，効率的なタスク処理が実現できることを示唆するものである．

今後の課題として以下が挙げられる．

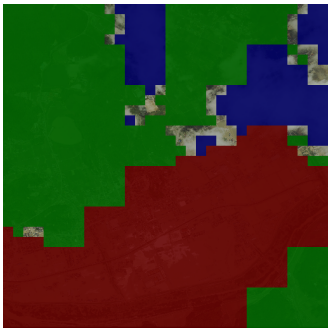
タスク結果品質の定義　本研究での AI ワーカの精度は，人間ワーカのタスク結果と同じであるタスク結果の数の比率である．ここには，AI ワーカが人間ワーカと同じように振る舞いことを良しとする前提がある．つまり，AI ワーカが人間ワーカよりも本質的に正しい回答をするような状況下では，AI ワーカの能力を適切に適切に測ることが出来ない．この問題に対処するための現実的な方法は，人間ワーカから得られるタスク結果をできるだけ正確にすることである [7, 31]．４章では，クラウドソーシングにおける代表的な品質管理手法である多数決と CTA を組み合わせるアルゴリズムを提案する．５章では，個々の人間ワーカからより正確なタスク結果を得るためのタスク設計について述べる．

評価するタスククラスタの選択　本研究では，AI ワーカから得られた全てのタスククラスタを評価の対象としたが，必ずしも全てのタスククラスタを評価する必要はない．そこで，タスククラスタの候補から評価が必要なタスククラスタを選択することが有効であると考えられる．例えばプラットフォーム側で動作するクラスタリングアルゴリズムや依頼者やドメインの専門家の経験則に基づくルールで得られたタスククラスタと AI ワーカのタスククラスタの重複などに着目して採用できる見込みがあるタスククラスタかどうかを判断できる可能性がある．このようなアイデアに基づいて割り当てアルゴリズムを拡張することで，より高速にタスク割り当てを決定することができる．

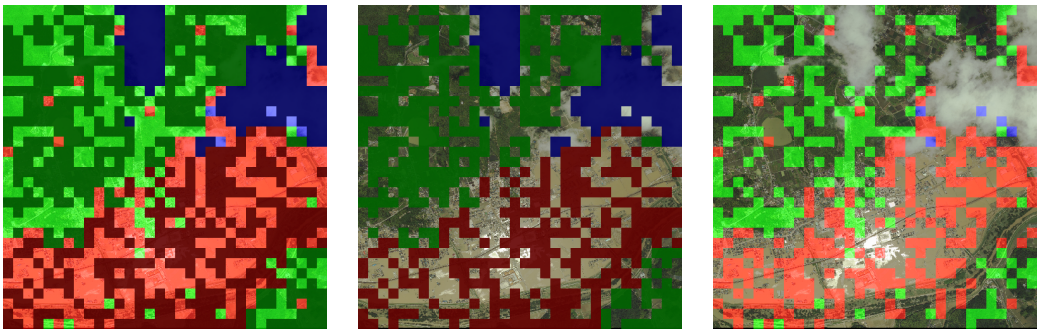
タスククラスタの効率的な評価　本章で提案したタスク割り当てアルゴリズムは，特定の AI ワーカには依存せず，任意の AI ワーカを同時に扱えるという点でスケーラブルである．しかし，評価対象のタスククラスタや各タスククラスタに含まれるタスク数が膨大である場合でも効率的に処理できるわけではない．この問題に対処するためには，タスククラスタを並列で評価することや，前述の方法などに基づいて評価対象のタスククラスタの絞り込みが必要となる．このようなアイデアに基づいてタスククラスタを拡張することは重要な今後の課題である．

(A) Original Image

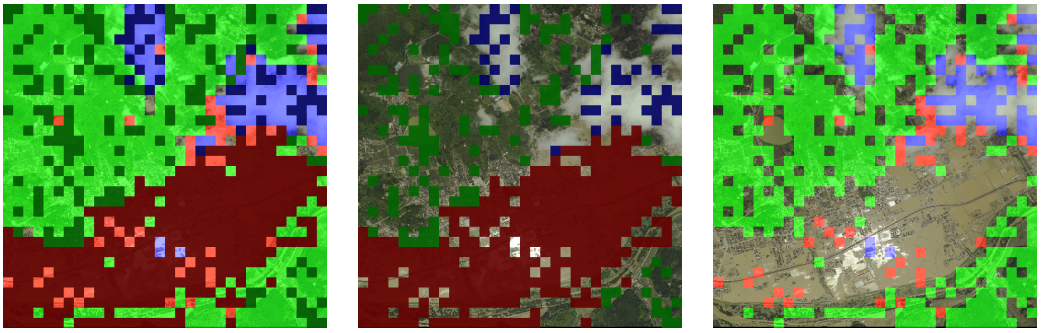
(B) Ground Truth



(C) Labels by ALA ($q = 0.85$)



(D) Labels by GTA ($q = 0.85$)



Human+AI　　Assigned to humans　　Assigned to AI workers

表 3.2: 水害被害判定タスクにおける各割り当てアルゴリズムで得られたタスク結果の詳細：(A) 浸水箇所を含む航空写真（B) Amazon Mechanical Turk の人間ワーカから得た正解とみなすラベル (C)ALA で得られたタスク結果（D)GTA で得られたタスク結果（研究業績 2-1 を改変）

第4章　人間+AIクラウドの相互作用に基づくタスク割り当て手法の拡張

この章では，群衆によって開発された不特定多数の AI プログラムを，人間のクラウドワーカと同様に扱う"人間+AI Crowd"世界でのクラウドソーシングにおいて，必ずしも正答するとは限らない人間および AI ワーカの相互作用により，低コストで高品質なタスク結果を実現するタスク割り当て問題を扱う．この"人間+AI Crowd"世界では，人間ワーカのタスク結果は単なる成果物としてだけでなく，複数の AI ワーカの学習と評価に利用される．そして，最初は人間によって行われていた作業のうち，AI 化が可能な部分が徐々に自動的に AI による作業に置き換えられる．したがって，人間ワーカから得られるタスク結果品質は特に重要であるが，現実のクラウドソーシングでは，様々な理由で人間ワーカからのタスク結果が不正確な可能性があり，これにより最終的なタスク結果品質の低下を引き起こす．そのため，多数決などの集約手法を適用することが一般的であるが，このコストをできるだけ削減したい．本論文では，人間ワーカと AI ワーカの回答の不一致に着目し，人間ワーカの追加タスクの必要性を判定することで，人間ワーカのタスク数の増加を抑えながらタスク結果品質を改善する手法を提案する．ベンチマークデータセットを用いた実験結果から，不確実な人間ワーカと AI ワーカが相互にタスク結果を共有することで，タスク結果品質を改善しながら効率的にタスクを処理する仕組みが構築可能であることを示す．

4.1　はじめに

クラウドソーシングは不特定多数の人間のワーカに作業を依頼することで，依頼者の課題を解決するための有効な手段である．クラウドソーシングの課題として，実際に人間のワーカが作業することから，得られた作業結果に誤りを含む可能性があることや，作業を依頼可能な人間ワーカは限られているため，作業効率には限度があることが挙げられる．

一方で近年，AI プログラムの作成を外部協力者にクラウドソーシングする試みが普及しており，代表的な例として Kaggle[1]や AIcrowd[2]，SIGNATE[3]が挙げられる．

[1] https://www.kaggle.com
[2] https://www.aicrowd.com
[3] https://signate.jp

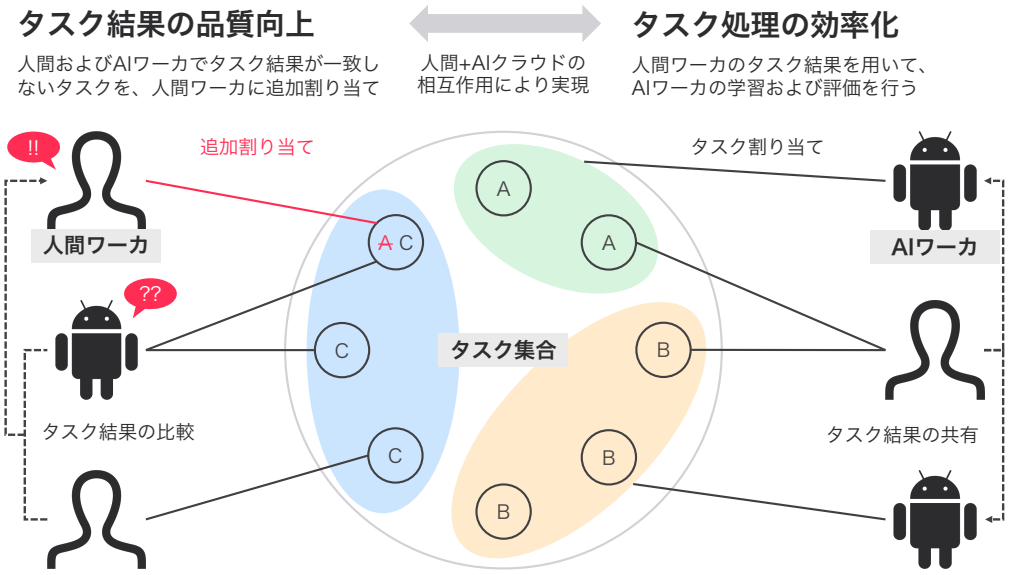


図 4.1: 本論文では，人間ワーカと AI ワーカの相互作用を設計することで，全体的なタスク結果品質を改善しながらタスクを処理する仕組みを提案する．(研究業績 1-1 より転載)

このような仕組みを活用すれば，AI 技術に精通していない依頼者であっても最先端の AI 技術の恩恵を受けられる．

しかしながら，AI プログラムの作成を依頼し，依頼者の課題解決に適用するまでのプロセスには手動での試行錯誤を伴うことが多い．例として，クラウドソーシングと機械学習モデルを組み合わせて，依頼者が要求する品質で 10,000 枚の画像を 10 クラス分類することを考える．依頼者はまず，クラウドソーシングを用いて一部の画像にラベルを付与して，機械学習モデルを作成するためのデータセットを構築する．次に，作成したデータセットを用いて，機械学習モデルの作成を Kaggle 等のプラットフォームで依頼する．このような手順を踏むことで，依頼者は機械学習モデルを入手できる．しかし，その機械学習モデルの出力が依頼者の要求する精度を満たすとは限らない．したがって，依頼者は学習およびテストデータを拡充した上で機械学習モデルの開発を再依頼するか，AI の利用を諦めて全ての画像の分類をクラウドソーシングで行うといった意思決定が求められる．このような工程を経て，人間による処理と計算機による処理を適切に組み合わせることは，専門家であっても容易ではない．

本研究はこの問題について，AI プログラムが利用するモデルやアルゴリズムの情報を使わず，ブラックボックスな"AI ワーカ"としてモデル化して扱う"人間+AI Crowd"アプローチを提案する．まず，先行研究で「人間ワーカは正しい」という前提で，依頼者の要求精度を満たすような人間ワーカおよび AI ワーカへのタスク割り当て問題を提起し，この問題に対するタスク割り当てアルゴリズムを提案した [55, 56]．このアルゴリズムは，人間ワーカからのタスク結果を基準として AI ワーカの性能を統

計的検定により評価する．AI ワーカの出力の全体を評価するのではなく，AI ワーカが出力するラベルの種類ごと（タスククラスタ，3.2 節で説明）に評価し，より多くのタスクを AI ワーカに割り当てる．

先行研究 [55, 56] のアルゴリズムでは，人間ワーカのタスク結果を成果物の一部として利用するだけでなく，AI ワーカの学習や評価に用いる．そのため，ワーカは常に正しいという前提を置いているが，現実には人間ワーカから得られるタスク結果が正しいとは限らない．クラウドソーシングにおける品質管理の研究では，異なる複数の人間ワーカに対して同一のタスクを割り当てて，それらの結果を統合して，より正確なタスク結果を得る多数決などの集約に基づく方法が広く用いられている [57]．これにより，全データに対して複数のタスク結果の集約を適用すれば，人間ワーカから得られるタスク結果品質の改善が期待できるが，人間ワーカへ依頼するタスク数が膨大になることが懸念される．

そこで本論文では，人間ワーカと AI ワーカの結果を相互に比較して，全体的なタスク結果の品質向上を行う仕組みを提案する（図 4.1）．本論文では，正解のあるタスクを対象とし，品質が高い結果とは，その正解率が高いことを意味する．また，品質が高いワーカとは，そのタスクに対して高品質が期待されるワーカである．提案手法では，人間ワーカと AI ワーカのタスク結果の不一致に基づいて，人間ワーカに追加のタスク割り当てを行うことで，人間ワーカから得られるタスク結果の品質（正解率）を向上させる．高品質な人間ワーカのタスク結果を AI ワーカの学習や評価のデータセットとして用いることで，より多くのタスクを AI ワーカに割り当てることを目指す．実験結果から，提案手法が全てのタスクに多数決を適用する手法と大きく変わらない品質のまま，人間ワーカへの追加割り当て数を大幅に削減できる可能性が示唆された．本手法は理論的に最悪のケースでも既存手法と同等の結果をもたらすため，実用的であると考えられる．

本研究の貢献は次の通りである:

- 先行研究 [55, 56] で提案した人間＋ AI クラウドタスク割り当て問題のアルゴリズムの性能を，人間ワーカのタスク結果が不正確な状況設定で評価した．
- 既存のタスク割り当てアルゴリズム [55, 56] と多数決を組み合わせることで，人間ワーカのタスク結果が不正確でも要求精度を満たすタスク割り当てが可能であるが，人間ワーカへのタスク割り当てを削減できる余地があることを示した．
- 人間ワーカと AI ワーカの不一致に基づいた人間ワーカへの追加タスク割り当てと AI ワーカの評価を組み合わせたアルゴリズムを提案した．実験結果から，提案アルゴリズムが全タスクに多数決を適用する手法と比較して，人間ワーカへの追加割り当て数を大幅に削減しながら，多数決を適用する手法と大きく変わらない品質の結果をもたらすケースがあることを示した．本手法での最悪の

ケースは，最大限人間に割り当てる場合だが，その場合も既存手法と同等の結果になる．

4.2 提案手法

4.2.1 問題設定

本研究で扱う問題は，全体的なタスク結果品質が要求精度 q を満たすようなタスク割り当て S を求めることである．ここで，分類タスクの集合 T，AI ワーカの集合 W，要求精度 q および，h の確率で正解を返す人間ワーカが与えられる．ただし，同一のタスクに対して人間ワーカを v 回割り当て，その結果の多数決を人間ワーカからのラベルとして利用することを許す．

4.2.2 本研究のアプローチ

本論文で提案するタスク割り当てアルゴリズムを説明する前に，アルゴリズムを構成する 2 つの要素について説明する．

人間ワーカへの追加タスク割り当て

本論文の問題設定では，人間ワーカに割り当てたタスクの結果を最終的なタスク結果および AI ワーカの学習・評価のために用いる．そのため，人間ワーカから得られるタスク結果品質を高めることは，全体的なタスク結果品質の向上に繋がる．ここで，タスク結果品質とは具体的にはタスク結果の精度とする．

人間ワーカから得られるタスク結果品質の管理手法として最も単純なのは，人間ワーカに割り当てる全タスクへの多数決の適用である．多数決により一部の回答が不正確であっても全体として品質の高いタスク結果を得ることができる．図 4.2 は，2 値分類における，多数決を行う人数（v）と個々の人間ワーカの正答率（h）の組み合わせにおいて，多数決結果の正答率のシミュレーション結果を図示したものである．多値分類で間違いパターンが偏らない場合などでは多数決が有効に働く閾値は 0.5 ではないが，傾向は同様になる．つまり，ワーカの平均正解率が閾値を超えている場合に多数決は有効であり，より多くのワーカが参加すれば正解率が上がる．本論文の提案アルゴリズムは，このような多数決によるタスク結果品質の改善が有効なケースを対象にする．

一方，人間ワーカに割り当てる全タスクへの多数決は，タスク数が膨大な状況では現実的でない．そこで，本論文では AI ワーカを用いて多数決を必要とするタスクを選択することを提案する．あるタスクに対して，複数の人間ワーカを割り当てて多数決を適用することを追加割り当てと呼ぶ．

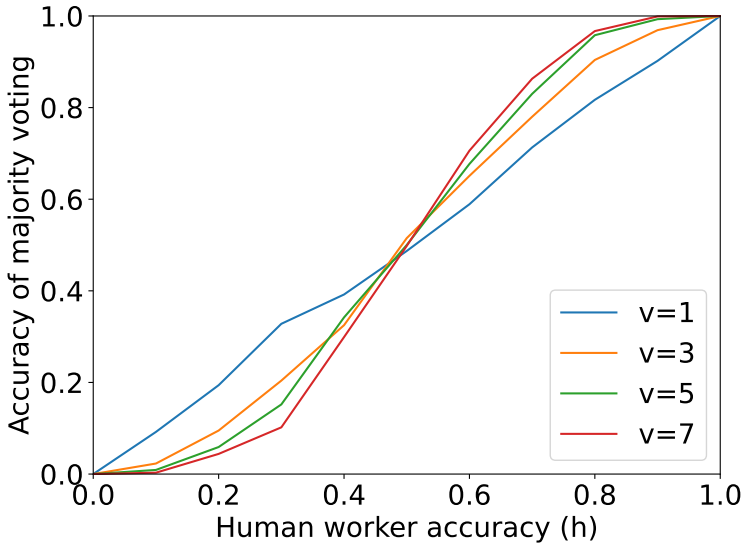


図 4.2: 個々のワーカの正答率と多数決する人数ごとの正答率の関係.（研究業績 1-1 より転載）

追加割り当てを行うタスクを選択するにあたり，人間ワーカと AI ワーカのタスク結果に不一致が生じたタスクに注目する．不一致であるは，人間ワーカもしくは AI ワーカのいずれかもしくは両方が誤りである可能性が高いので，(1) まずは人間ワーカのタスク結果品質を改善し，(2) そのタスク結果を用いて AI ワーカの訓練と評価を行う．この処理を繰り返すことで，より正確に全タスクを処理することを目指す．

活用と探索

本研究の問題設定ではより多くのタスクを AI ワーカに割り当てることが求められるが，一方で AI ワーカの訓練と評価を正確に行うために人間ワーカからのタスク結果品質が重要である．しかし，多数決による品質管理には十分な票数が必要である．

そこで，人間ワーカへの追加割り当てを探索，AI ワーカへのタスク割り当てを活用とし，両者の処理をバランス良く実行することを考える．ここで，活用と探索の処理の使い分けを制御するパラメータ ϵ を導入する．ただし，$0 \leq \epsilon \leq 1$ である．$[0, 1]$ の範囲の一様分布 U から乱数 $u \sim U$ を取得する．$\epsilon < u$ の場合，探索を行う．具体的には，4.2.2 節で説明した通り，人間ワーカと AI ワーカのタスク結果の不一致に基づき，人間ワーカに対して追加タスク割り当てを行うことで，人間ワーカから得られるタスク結果品質を向上させる．一方で $\epsilon \geqq u$ の場合，活用を行う．具体的には，タスククラスタ集合 C を評価して，AI ワーカに対してタスク割り当てを行う．

全体としてより多くのタスク結果を AI ワーカから採用するために，タスク割り当ての進行に応じて ϵ を動的に変動させることを考える．人間ワーカへの割り当て

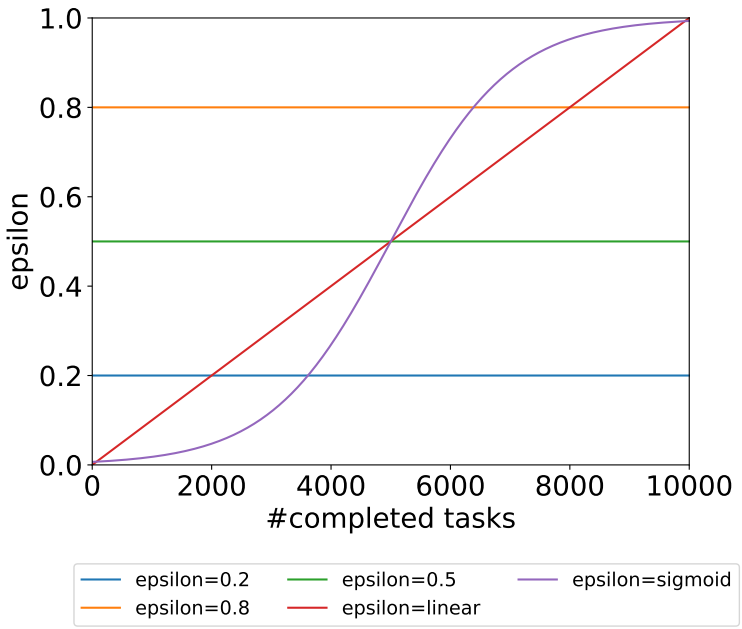


図 4.3: 総タスク数が 10000 の場合の，タスク割り当ての進行と各探索方策における ϵ の関係.（研究業績 1-1 より転載）

タスク数が少ない状態では，AI ワーカからのタスク結果を採用できる可能性は低いので，優先的に探索を実行することが有効である．しかし，人間ワーカへの割り当て済みタスク数が増えるにつれ，AI ワーカの性能が十分となる可能性が高くなるので，活用を優先して行うことが有効である．式 4.1 と式 4.2 に，タスク割り当ての進行に応じて値が変動する ϵ_{linear} と $\epsilon_{sigmoid}$ を示す．

$$\epsilon_{linear} = \frac{|t \mid t \in T, ans(t) = (\emptyset, \emptyset)|}{|T|} \tag{4.1}$$

$$\epsilon_{sigmoid} = \frac{1}{2}(1 + \tanh(\frac{1}{2}\alpha\epsilon_{linear})) \tag{4.2}$$

ただし，実験では $\alpha = 10$ を用いた．図 4.3 に，完了タスク数の増加に伴う ϵ の変化と，実験で比較する $\epsilon = 0.2, 0.5, 0.8$ を示す．

4.2.3 提案アルゴリズム

CTA では，人間ワーカから得られるタスク結果が正しいと仮定して，人間ワーカのタスク結果を AI ワーカの学習と評価に用いる．そのため，本論文の実験結果が示すように，人間ワーカのタスク結果に誤りが含まれる場合に，要求精度を満たす割り当てを得られない．本論文では，人間ワーカのタスク結果が正しいとは限らない状況を考慮したアルゴリズムを提案する．

Algorithm 4 Intaractive Clusterwise Test-based Assignment（ICTA）

Require: A set T of tasks, a set W of AI workers, the accuracy requirement q, the significance level α, and the intaractive threshold ϵ.

Ensure: A sequence of (task, worker) pairs.

1: **for all** $t \in T$ s.t. $ans(t) = (\emptyset, \emptyset)$ in a random order **do**
2: $a' \leftarrow task\_result(assign(t, `h'))$
3: update ans so that $ans(t) = (a', `h')$
4: **if** $\epsilon < random()$ **then**
5: let Q be a multiset
6: **for all** $T_k \in C$ **do**
7: $\hat{a} = \underset{a \in A}{arg\ max} |\{t \mid t \in T_k, ans(t) = (a, `h')\}|$
8: $Q \leftarrow Q \cup \{t \mid t \in T_k, a \in A, ans(t) = (a, `h'), a \neq \hat{a}\}$
9: **end for**
10: let t' be the most frequent element in Q
11: $a' \leftarrow majority\_vote([ans(t'), task\_result(assign(t', `h'))])$
12: update ans so that $ans(t') = (a', `h`)$
13: **else**
14: **for all** $T_{w,j} \in C$ **do**
15: $\hat{a} = \underset{a \in A}{arg\ max} |\{t \mid t \in T_{w,j}, ans(t) = (a, `h')\}|$
16: **if** $statistical\_test(T_{w,j}, \hat{a}, q, \alpha)$ **then**
17: for $t' \in T_{w,j}$ s.t. $ans(t') = (\emptyset, \emptyset)$, $assign(t', w)$ and update ans so that $ans(t') = (\hat{a}, w)$
18: **end if**
19: **end for**
20: **end if**
21: **end for**

提案アルゴリズムである Interactive Clusterwise Test-based Assignment（アルゴリズム 4）について説明する．ICTA では，人間ワーカと AI ワーカのタスク結果の不一致に基づいて，人間ワーカに追加タスク割り当てを行うことで，人間ワーカから得られるタスク結果品質の向上を図り，高品質な訓練およびテストデータによって AI ワーカを活用する．これにより，人間ワーカと AI ワーカのタスク結果品質を相互に改善しながらタスク割り当てを行う．4 行目で ϵ と乱数を比較し，探索（5 - 12 行目）を行うか，活用（14 - 19 行目）を行うかを決定する．6 - 9 行目では全てのタスククラスタについて，人間ワーカと AI ワーカのタスク結果が不一致であるタスクを列挙する．10 行目で最も不一致の頻度が高いタスクを選択し，11 行目でそのタスクを $v - 1$ 名の追加の人間ワーカに割り当てた上で $majority\_vote$ 関数により多数決を行う．12 行目では選択したタスクの結果を多数決の結果で更新する．

4.3 評価実験

4.3.1 実験設定

くずし字データセットである KMNIST[58] を用いて，10 クラス分類タスクを作成した．訓練データから 10,000 件の画像とラベルのペアを無作為抽出し，タスク集合として用いた．

AI ワーカ集合として，scikit-learn 0.23.1 に実装されている機械学習モデルの中から，次のクラス名で定義されたモデルを初期パラメータの設定で用いた: MLPClassifier，ExtraTreeClassifier，LogisticRegression，KMeans，DecisionTreeClassifier，SVC.

実験では，既存手法の CTA と提案手法の ICTA を比較する．統計的検定には二項検定を利用し，有意水準は 5%とした．実験で使用したソースコードは GitHub リポジトリに公開されている[4].

4.3.2 結果

以下に示すパラメータで，10 試行ずつ行った実験結果を示す.

- 多数決の人数 $v \in \{1, 3, 5, 7\}$
- 人間ワーカの正答率 $h \in \{0.8, 0.9, 1.0\}$
- 要求精度 $q \in \{0.8, 0.85, 0.9, 0.95\}$
- $\epsilon \in \{0.2, 0.5, 0.8, \epsilon_{linear}, \epsilon_{sigmoid}\}$
- 人間ワーカへの追加割り当ての基準: 人間ワーカと AI ワーカが一致，人間ワーカと AI ワーカが不一致，ランダム選択

人間ワーカの正答率を変えた実験

人間ワーカが正答を返すとは限らない設定 ($h \in \{0.8, 0.9, 1.0\}$) における CTA の振る舞いを明らかにするための比較実験を行った (図 4.4)．上段が人間ワーカによるタスク結果数と完了タスク数の関係を，下段が全体的なタスク結果品質および AI ワーカが処理した部分の精度と要求精度パラメータの関係を表している．エラーバーは標準偏差を表す．上段の図において，横軸に比例した総タスク数の増加は人間ワーカによるタスク処理を意味し，一方で横軸の値の変動を伴わない総タスク数の増加は AI ワーカによるタスク処理を意味する．下段の図において，全体的なタスク結果が，緑色の点線で示される要求精度を下回っている場合，そのタスク割り当ては依

[4] https://github.com/crowd4u/HACTAP-Framework

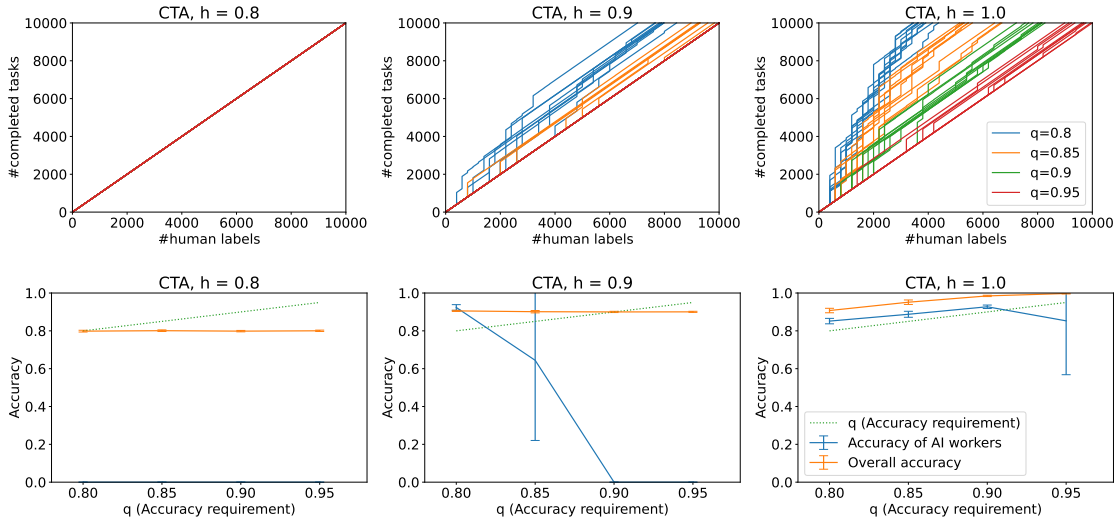


図 4.4: 異なる人間ワーカの正答率での実験結果: 人間ワーカへのタスク割り当て数と完了タスク数の関係 (上)，要求精度と実際のタスク結果の精度の関係 (下).（研究業績 1-1 より転載）

頼者の要求を満たしていない．青色で示される AI ワーカが処理した部分のタスク結果は要求精度を上回っている必要はないが，アルゴリズムの評価のため採用した AI ワーカの実際の品質を示したものである．

実験結果から，h が低いほど要求精度を達成できる割合が低下する．具体的には，q が h を上回る条件では要求精度を満たさなかった．$h = 0.9$ における $q = 0.8$ の設定では，AI ワーカが処理した部分のタスク結果品質が h を上回った．これは，ノイズを含むデータを学習した AI ワーカが真の正解に近い分類を行ったことを意味する．一方で，h が高いと人間ワーカへの割り当てが少ない状況で AI ワーカへの割り当てが行われる傾向が見られた．

多数決の人数を変えた実験

CTA で人間ワーカにタスク割り当てる際に多数決を行う設定で，アルゴリズムの挙動を明らかにする実験を行った．実験では人間ワーカの正答率を $h = 0.8$ とし，多数決の人数 v を変化させた．$v \in \{3, 5, 7\}$ の設定における実験結果を図 4.5 に示す．

実験結果から，v が高いほど，要求精度を満たせる頻度が増加した．$v = 3$ の設定では，$q = 0.95$ のときに要求精度を達成することが出来なかったが，$v = 5, 7$ の設定では全ての設定で要求精度を満たした．図 4.4 の $h = 0.8$ の結果は $v = 1$ の設定と同等であり，一連の結果から要求精度を満たせる頻度は v の増加に伴い向上したと言える．この結果は図 4.2 の関係からも裏付けられる．

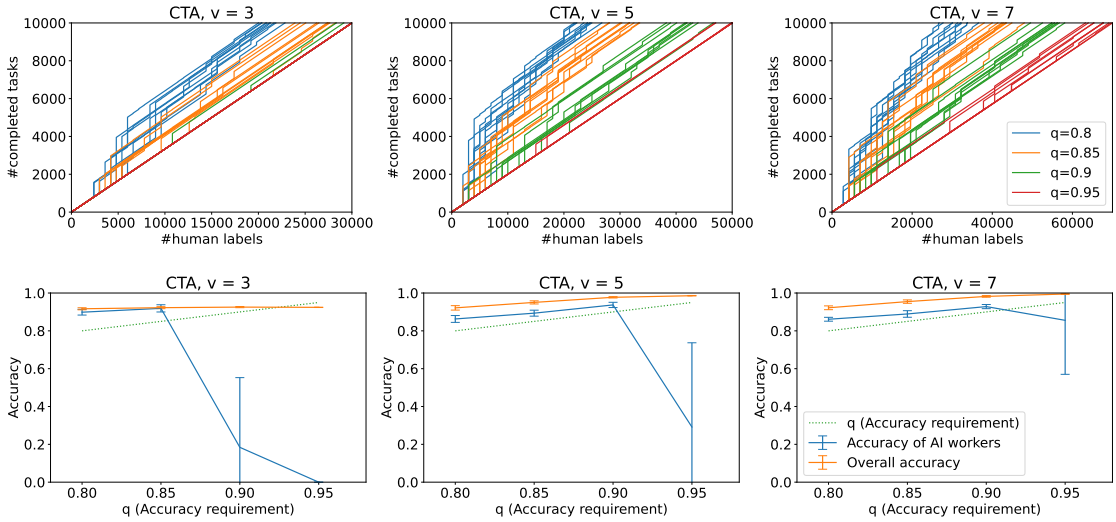


図 4.5: 多数決の人数を変えた実験の結果: 人間ワーカへのタスク割り当て数と完了タスク数の関係 (上)，要求精度と実際のタスク結果の精度の関係 (下).（研究業績 1-1 より転載）

活用と探索のパラメータを変えた実験

ϵ が AI ワーカへのタスク割り当て数および全体的なタスク結果品質に与える影響を明らかにする．実験では，人間ワーカの正答率を $h = 0.8$，各タスクの人間ワーカへの最大割り当て数 $v = 5$ とし，活用と探索のパラメータごと ($\epsilon \in \{0.2, 0.5, 0.8, \epsilon_{linear}, \epsilon_{sigmoid}\}$) の結果を比較した（図 4.6）．

$\epsilon = 0.2, 0.5$ の設定では，全ての設定で要求精度を満たすことができた．AI ワーカへのタスク割り当て数を図 4.5 における $v = 5$ の設定と比較すると，$\epsilon = 0.2$ の方がより多くのタスクを AI ワーカに割り当てていることが分かる．一方で，$\epsilon = 0.8$ の設定では $q = 0.8, 0.85$ の場合のみ，要求精度を満たすことができた．この結果から，ϵ の適切な調整により，全タスクに人間ワーカへの追加割り当てを行うことなく，全体として十分な品質をもたらすタスク割り当てが可能であることが示唆された．

要求精度 $q = 0.8, 0.85$ の設定において，$\epsilon = 0.8$ では AI ワーカから得られたタスク結果の品質が要求精度を下回ったが，ϵ をタスク割り当ての進行に伴って変更する $\epsilon_{linear}, \epsilon_{sigmoid}$ では上回った．さらに，$q = 0.9$ において全体のタスク結果品質が要求精度を上回った．この結果は，人間ワーカへの追加割り当ての回数は同程度であったとしても，追加割り当てを行うタイミングを調整すればより多くのタスクを AI ワーカに割り当てられることを示している．しかし，$q = 0.95$ の設定では要求精度を満たせず，追加割り当ての許容回数や ϵ の調整手法に改良の余地がある．

人間ワーカへの追加割り当ての戦略を変えた実験

人間ワーカと AI ワーカのタスク結果の比較方法が AI ワーカへのタスク割り当て数に与える影響を明らかにするための実験を行った．実験では，人間ワーカの正答率を $h = 0.8$，各タスクの人間ワーカへの最大割り当て数 $v = 5$, $\epsilon \in \{0.2, \epsilon_{linear}, \epsilon_{sigmoid}\}$ とし，一致およびランダム選択の条件を比較した．

追加割り当ての基準を人間と AI ワーカの一致とした場合の結果を図 4.7 に示す．不一致に基づく場合の結果である図 4.6 では AI ワーカが処理したタスクの精度が q を上回っていた $q = 0.9$ について，一致に基づく設定では q を満たさなかった．ランダム選択の結果を図 4.8 に示す．不一致選択の設定では AI ワーカの精度が要求精度を上回った $\epsilon_{sigmoid}, q = 0.9$ の設定において，ランダム選択では割り当てられた AI ワーカの精度の平均値が要求精度を下回った．このことから，人間ワーカへの追加割り当てを行うタスクを選ぶために，人間ワーカと AI ワーカの不一致に基づく手法が有効であった．この結果は，不一致に基づいて追加割り当てを行うことが，AI ワーカの訓練および評価のためのデータ品質改善に有効であり，その結果，AI ワーカの各タスククラスタが採用出来たからであると考えられる．

4.3.3 考察

人間ワーカの正答率を変えた実験では，h が低い設定では CTA で要求精度を満たす割り当てを得られなかった．これは，CTA が人間ワーカのタスク結果が正しいことを前提とするアルゴリズムだからである．このことから，人間ワーカから得られるタスク結果品質を高めることが重要であると言える．

多数決の人数を変えた実験では，v を高く設定するほど要求精度を満たせる割合が向上した．人間ワーカに割り当てる全タスクへの多数決が，前述の課題に有効である．

提案手法において，活用と探索パラメータを変えた実験では，ϵ の適切な設定により，要求精度を満たしながら人間ワーカへの追加タスク割り当て数を削減できることを示唆する結果が得られた．タスクの進行に伴って ϵ を変動させた設定は，固定値を用いる設定よりも人間ワーカへの追加割り当て数が少なかったことから，タスクの進行状況やその他の情報に基づく ϵ の調節が有効であると言える．一方で，タスクの進行に伴って ϵ を変動させた設定は，固定値を用いる設定よりも，最初に AI ワーカにタスクを割り当てるまでに多くの人間ワーカ割り当てを必要とする傾向が見られた．これは，タスク割り当ての序盤に AI ワーカの評価が行われる確率が低いからである．特に要求精度が低い場合に，要求精度を満たす AI ワーカが存在するにも関わらず，探索により結果として人間ワーカへのタスク割り当て数が増大すると予想される．このような観点からも ϵ の動的な変更には改善の余地がある．

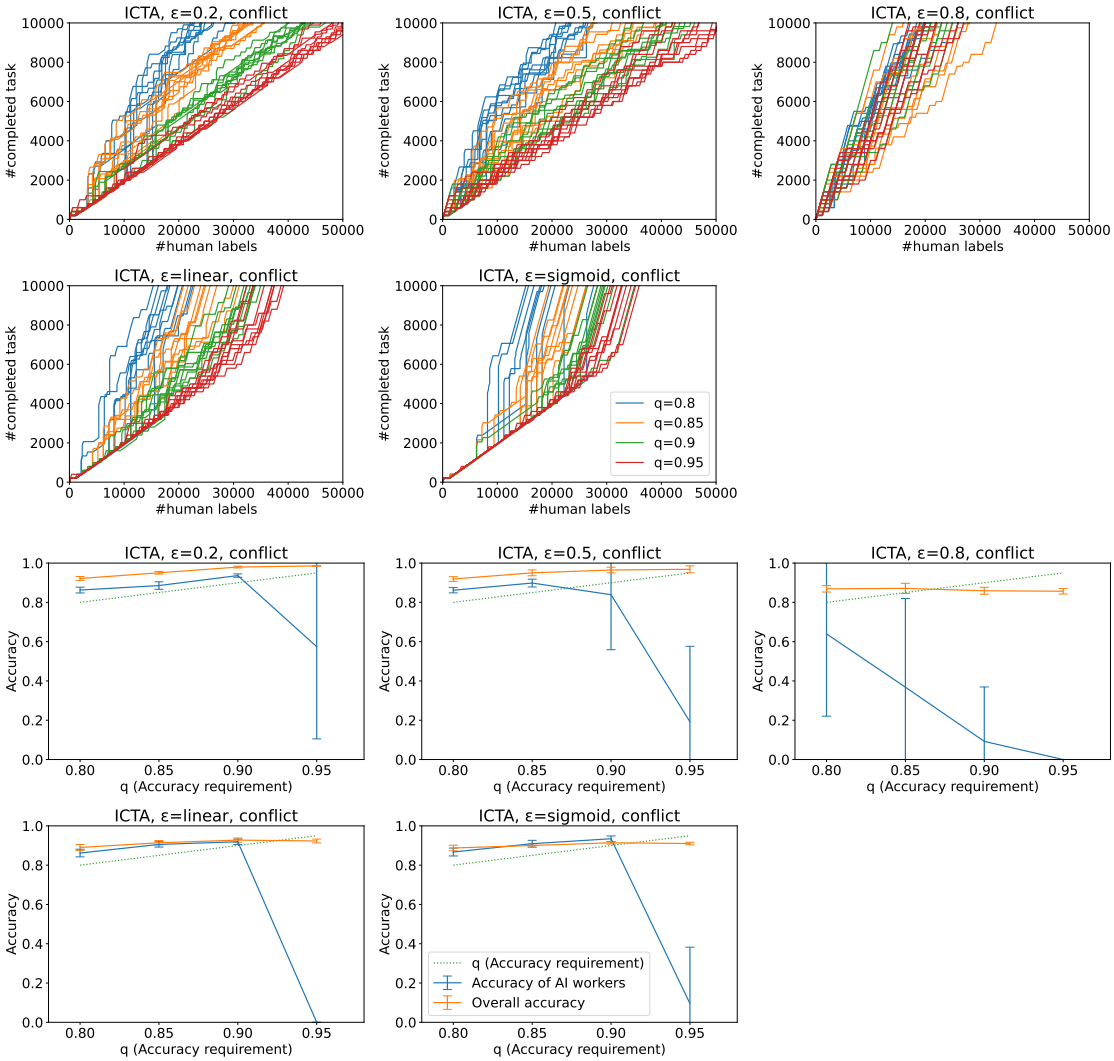


図 4.6: 活用と探索のパラメータを変えた実験の結果: 人間ワーカへのタスク割り当て数と完了タスク数の関係 (上)，要求精度と実際のタスク結果の精度の関係 (下).（研究業績 1-1 より転載）

提案手法において，人間ワーカへの追加割り当て戦略を比較した実験では，人間ワーカと AI ワーカの不一致に基づく戦略が有効であった．本論文では，不一致であるタスクに基づいてのみ追加割り当てを行ったが，タスククラスタの評価結果に基づいて重みづけするなど，追加割り当てを行うタスクの選択手法には改善の余地がある．さらに，観測された人間ワーカと AI ワーカのタスク結果の不一致を，ϵ の動的な調整の根拠として用いることも提案手法の改良の方針として挙げられる．

4.4 まとめ

本研究では，人間ワーカと AI ワーカのタスク結果を相互に比較することで，人間ワーカからのタスク結果品質を改善しながら，より多くのタスクを AI ワーカに割り当てる手法について議論した．実験では，提案手法が人間ワーカと AI ワーカの不一

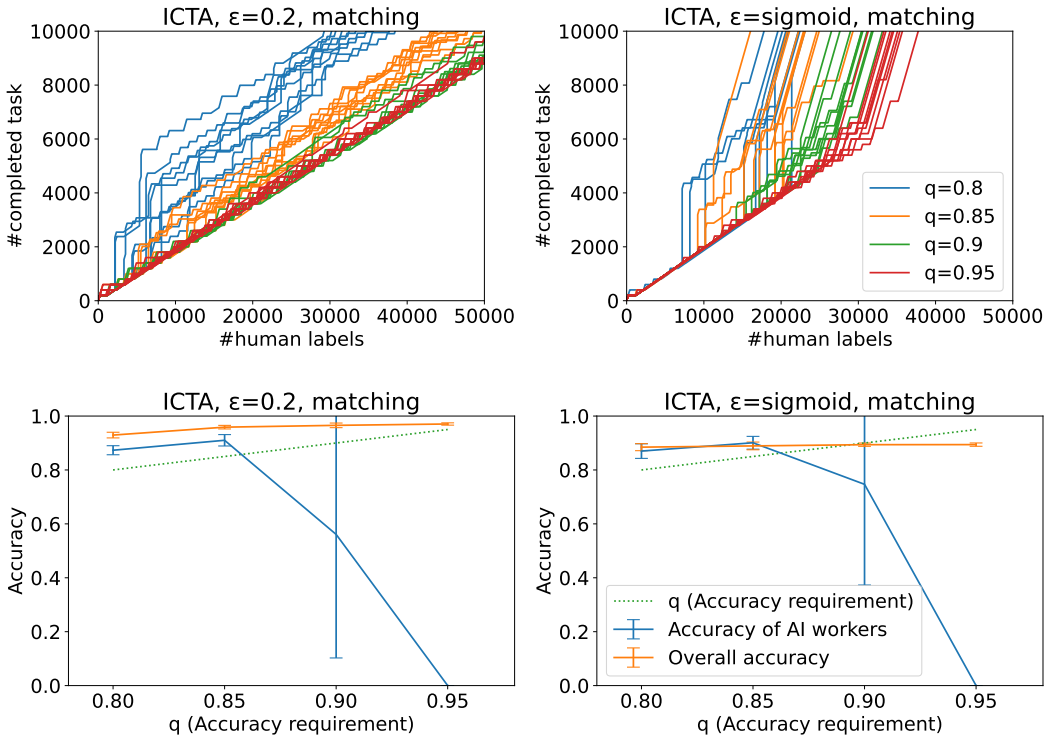


図 4.7: 追加割り当てを行うタスクを一致およびランダムに選択した実験の結果：人間ワーカへのタスク割り当て数と完了タスク数の関係 (上)，要求精度と実際のタスク結果の精度の関係 (下). (研究業績 1-1 より転載)

致に基づいて人間ワーカへの追加割り当てを行うことで，人間ワーカに割り当てる全タスクに多数決を適用するよりも人間ワーカへの割り当て数を大幅に削減しながら，多数決を適用する場合と同程度のタスク結果品質を得られるケースがあること示した．この結果から，人間ワーカのタスク結果が不正確であり，AI ワーカの学習およびその品質を統計的に評価することは難しい状況において，人間ワーカのタスク結果品質を高めることで AI ワーカをより活用できることが示唆された．

本論文では，人間ワーカが正答する確率が一定であると仮定し，確率を変化させた実験を行なった．今後の課題として，クラウドソーシング実験の経験則等に基づく合理的な確率分布を仮定したり，個々の人間ワーカの正答率が異なる状況を考慮した実験およびアルゴリズムの改良が挙げられる．

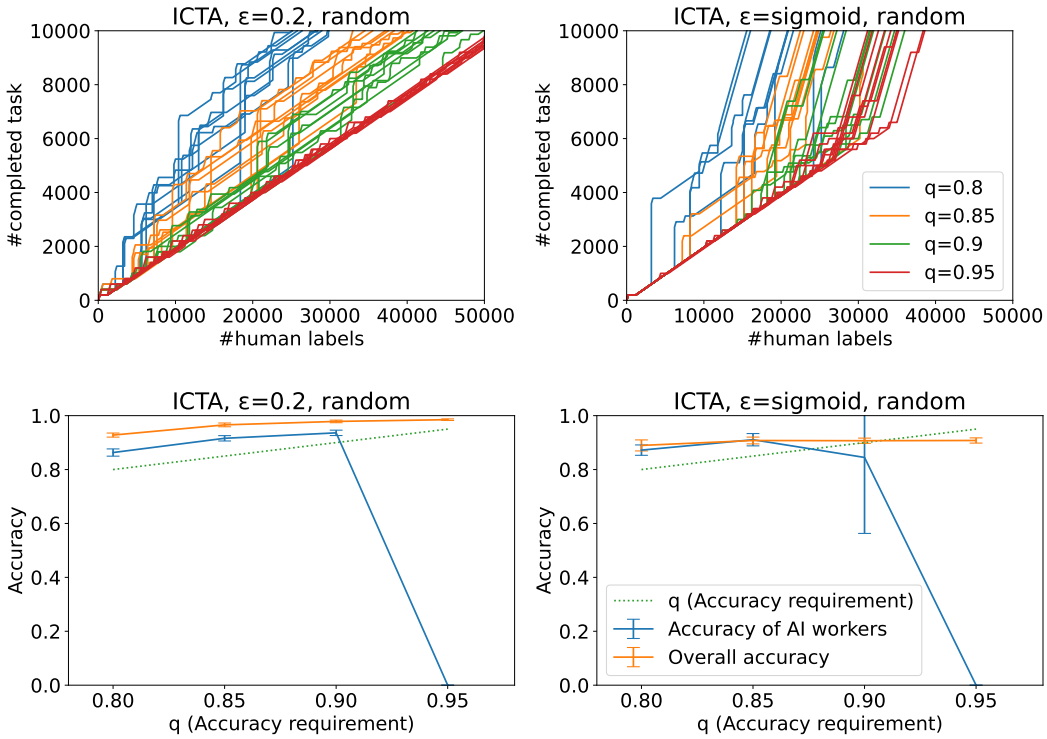


図 4.8: 追加割り当てを行うタスクを一致およびランダムに選択した実験の結果：人間ワーカへのタスク割り当て数と完了タスク数の関係 (上)，要求精度と実際のタスク結果の精度の関係 (下). (研究業績 1-1 より転載)

第5章　マイクロタスクにおける自己補正が人間ワーカにもたらす短期的・長期的効果の分析

5.1　目的

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．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．

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

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

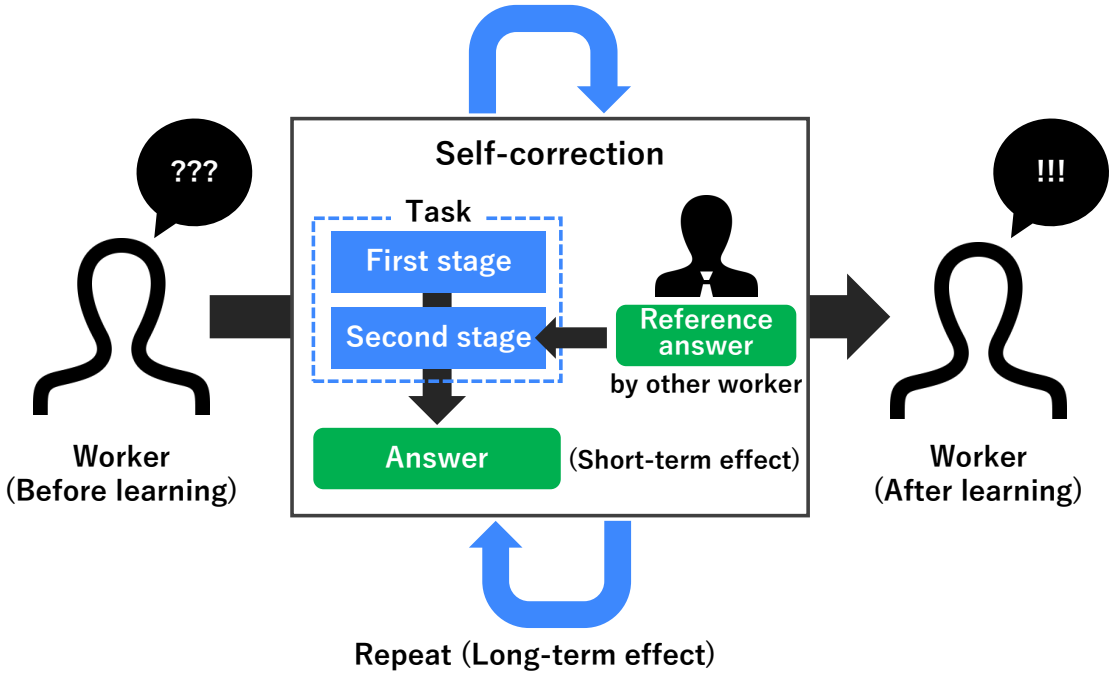


図 5.1: 本研究の概要（研究業績 1-2 より転載）

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．

しかし，自己補正が提案された論文では，自己補正の有効性についてシミュレーションによる評価のみが行われており，現実のクラウドソーシング環境においても同様の効果が得られるかは不明である．実際のクラウドソーシング環境においても自己補正が有効であるかは興味深い課題である．

さらに，自己補正がワーカに対して無自覚な学習をもたらすことが出来るかについても注目すべき点である．ワーカが自己補正タスクに取り組むことにより，その後の作業の品質を改善できるならば，正答が既知の課題を利用した訓練フェーズを導入することなく，ワーカの品質を改善できることを意味する．視覚的な作業を行う能力は，繰り返すことにより，すなわち知覚的な学習によりその速度や精度が改善されることが知られている [59]．知覚学習は意図せず無自覚に生じるものであり [60]，複数の研究が，視覚的な分類課題における知覚学習を報告している [61]．したがって，自己補正を繰り返すことにより，視覚的な分類タスクを行う作業者の分類精度が向上すると考えられる．

本研究では，現実のクラウドソーシング環境における自己補正の短期的および長期的効果を検証するための実験結果について報告する（図 5.1）．

本研究における主な実験結果は次のとおりである．

1. 自己補正によりタスク結果の品質改善（以降では自己補正の短期的効果と呼ぶ）は，現実のクラウドソーシング環境においても見られた．

2. 自己補正の第 2 段階で提示する参考回答は重要な要因であることが示された．さらに，より信頼性の高い参考回答を提示することで自己補正がもたらす効果を増大させることが示唆された．

3. ワーカが自己補正を繰り返すことにより，ワーカ自身の回答品質が改善された（以降では自己補正の長期的効果と呼ぶ）．この長期的な効果は，ワーカが同様のタスクに繰り返し取り組むことによって生じた知覚学習であると説明できる．この結果から，自己補正がもたらす効果が，以降の同様のタスクについても良い影響を与えることが示唆された．

4. 自己補正の長期的効果が，学習に用いた課題とは異なる課題においても正答率の改善をもたらすかを明らかにするために，自己補正を適用した学習タスクと評価タスクで異なる課題を用いる実験を検討した．その結果，学習の転移は見られなかった．

本研究の各実験では，成果物の品質にかかわらずワーカに対して定額の報酬を支払った．一方で，Shah[62] らは，自己補正に適用可能な報酬設定の手法を提案している．にもかかわらず，本研究で示す結果は Shah[62] の主張を支持するものであることから，自己補正は任意の報酬設定の手法が適用出来ないような環境においても有効であることが示唆された．Shah[62] が提案した手法を始めとする，様々な報酬設定の手法を現実にクラウドソーシングにて検証することは注目すべき点の１つである．

実験 1A では自己補正の短期的な効果と長期的な効果を鳥の画像分類課題で検討した．実験 1B では，実験 1A で見られた効果が，別の難易度の高いタスクにおいても同様の傾向であるかを，絵画の分類課題を用いて検討した．実験 2 では，実験 1A および実験 1B で見られた自己補正の長期的効果が，類似した別の課題においても品質の改善をもたらすかを，類似した複数の画像分類課題を組み合わせて検討した．

5.2 本研究のアプローチ

5.2.1 自己補正

この節では，Shah[62] らが提案した自己補正について，彼らの論文の貢献を説明する．

タスクの構成　一般的なクラウドソーシングサービスでは，ワーカは自身の誤りを発見して訂正する機会がない．しかし，多くのワーカ（スパムワーカなどを含まない）においては，誤りに気づく機会を提供することによって，ワーカが自らの回答を訂正することが出来ると考えられる．自己補正は，クラウドワーカらの成果物の品質を高めるためにのタスク設計である．自己補正では，ワーカは同じ質問に対して２回回答する機会が与えられる．１回目は，通常のクラウドソーシングタスクと同様に回答し，２回目では他者の回答を照らし合わせて回答を変更することが出来る．

報酬アルゴリズム　自己補正を適用したタスクでは，第２段階で他者の回答を考慮するのではなく，単に自身の回答を他者の回答で置き換えてしまうようなワーカの存在が想定される．そこで，Shah らは自己補正のための報酬アルゴリズムを提案した．彼らのアルゴリズムは，第１タスクで正答すると最も価値が高く，第２段階で他者の回答を支持すると低くなるような設定となっている．

シミュレーション　Shah らは，自己補正の有効性を明らかにするための，シミュレーションによる実験を行った．シミュレーションでは，自己補正を適用したタスクと通常のタスクを比較した．シミュレーションの結果は，自己補正を適用したタスクのほうが，最終的に得られる成果物の品質が高くなるというものである．彼らによれば，自己補正を適用することにより，成果物を用いるアプリケーション（例えば機械学習など）の品質が改善されるという．

5.2.2　実験環境

実験環境の構成　実験環境の概略を 5.2 に示す．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．

実験に参加するワーカはまず，Yahoo! クラウドソーシング上のタスク一覧画面から，本研究にて作成した募集ページを選択し，作業に参加する．Yahoo クラウドソーシングにおける作業画面には，Crowd4U 上のタスク画面へのリンクとリンク先のページにて実際の作業を行う説明があり，リンク先のページへ移動して実際の作業を行う．作業が完了すると，Crowd4U 上の作業完了ページへと画面遷移する．作業完了ページにはキーワードとトークンが，Yahoo! クラウドソーシング上の画面にてこれらを入力するという説明とともに表示されている．最後に，Yahoo! クラウドソーシング上の作業画面にてキーワードとトークンを入力し，作業を完了させることで，報酬を受け取る．

[2]https://crowdsourcing.yahoo.co.jp
[3]https://crowd4u.org

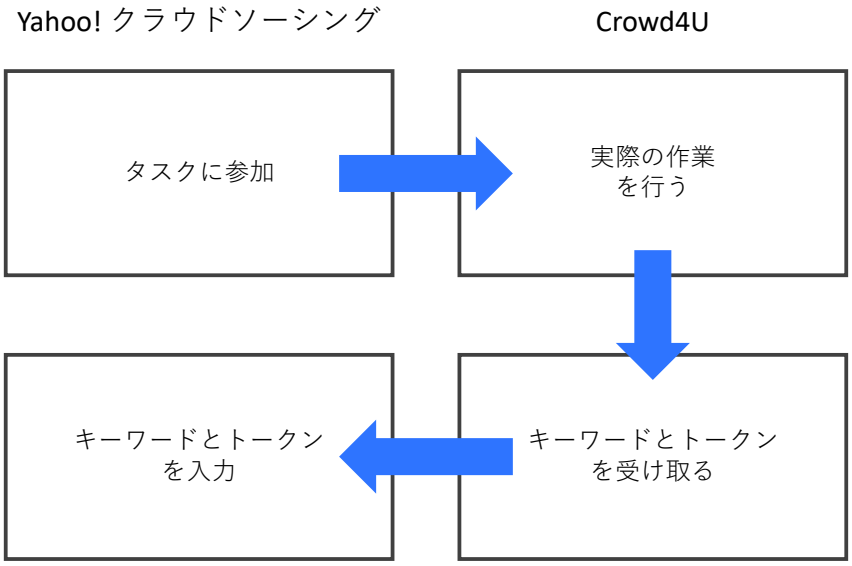


図 5.2: 実験環境の概要

Crowd4U 上での作業完了後に表示されるキーワードとトークンは最後まで作業に取り組み，報酬を受け取ったワーカを識別するためのものである．キーワードはすべてのワーカに共通な文字列であり，この文字列を得たワーカは作業を完了したとみなす．トークンはワーカごとに一意な文字列であり，個々のワーカを識別するためのものである．図 5.3 に Crowd4U での作業完了後に表示されるキーワードおよびトークンの表示画面の一例を示す．

クラウドワーカの募集　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．

各実験の設定の比較　図 5.4 に，本研究で取り組む各実験の全体としての目的を示す．全体の目的は，自己補正を現実のクラウドソーシングに適用した場合の，各自己補正タスクの正答率の改善（自己補正の短期的効果）と自己補正の繰り返しによるワーカ自身の正答率の改善（自己補正の長期的効果）が見られるかを検証することである．

表 5.1 に，各実験において設定が異なる点を示す．実験 1A では，自己補正における参考回答の有無が，自己補正の短期的効果および長期的効果にあたえる影響を検証する．実験では，実験に参加するワーカを 2 つのグループに分割し，片方のグルー

図 5.3: Crowd4U での作業完了後に表示されるキーワードおよびトークンの表示画面

プを自己補正における参考回答がありの条件に，もう一方のグループを参考回答が無しの条件に割り当てる．実験では鳥の画像分類課題を用いた．

実験 1B では，自己補正における参考回答の品質が，自己補正の短期的効果および長期的効果にあたえる影響を検証する．実験では，実験に参加するワーカを２つのグループに分割し，片方のグループを参考回答が常に正解の条件に，もう一方のグループを参考回答が選択肢からランダムに選んだ回答とする条件に割り当てる．実験では，絵画を提示してその作者を選択肢から選ぶ分類課題を用いた．

実験 2 では，自己補正における長期的効果が，学習に用いる課題と評価に用いる課題が異なる場合にも正答率の改善をもたらすか（学習の転移）を検証する．実験では，事前に平均正答率をクラウドソーシングにより測定したデータセットを複数組み合わせて，学習の転移がみられるかを明らかにする．比較する条件は，自己補正にて正解の参考回答を提示する場合と，自己補正を適用しない通常のタスクの場合を比較する．

5.3　実験 1（自己補正の短期的・長期的効果）

5.3.1　実験 1A（参考回答の有無の影響）

目的

クラウドソーシングにおいて，成果物の品質を改善することは重要な研究課題の１つである．Shah らが提案した自己補正は，作業に取り組むワーカに対して回答の機会を２度与えることでタスク結果の品質を改善する手法である．Shah らの論文では自己補正について，シミュレーションによる評価が行われたが，現実のクラウドワーカやタスクに対しても同様の傾向が見られるかは明らかでない．そこで本実験

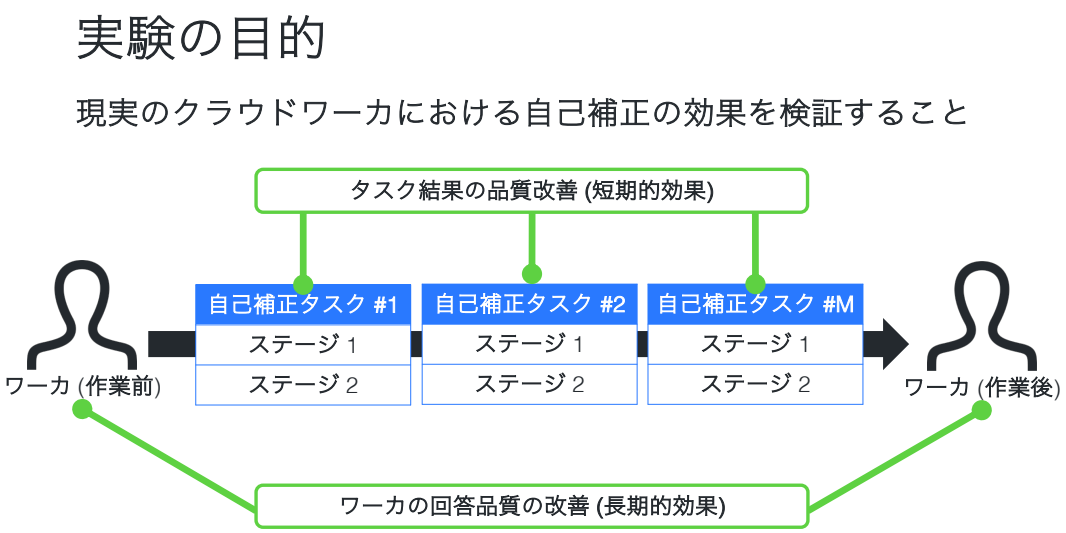


図 5.4: 実験全体での目的

では，現実のクラウドワーカに自己補正を適用することで，現実のクラウドワーカのタスク結果が改善されるかを検証することを目的とし，画像分類課題を行った．

自己補正を適用したタスクにより現実のワーカが自身の回答を改善するならば，実際のクラウドソーシングにおいて，リクエスタが自己補正の導入を検討出来るようになるだろう．自己補正は，既存のタスク結果の改善手法である，多数決のようなタスク結果の集約や，優れたワーカを見つけて優先してタスクを割り当てるなどの手法とも組み合わせることが容易であることから，多くの場面で活用できると考えられる．

現実のクラウドソーシングにおいて自己補正を導入する場合，どの様な参考回答を提示するかを検討する必要がある．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一般的であるため，正解を提示することは難しいため，すでに同様のタスクに回答したワーカの回答やその集約結果などを提示することになる．ただし，自己補正におけるタスク結果の品質改善において，２段階のタスクデザインと参考回答の提示の両者が重要な要因であるかは自明ではない．そこで本実験では，参考回答の有無により，タスク結果の品質改善にどの様な影響をもたらすかを検証する．

自己補正によるタスク結果の改善は，２段階目の回答の品質を改善することを目的とする手法である．しかし，ワーカが自己補正を繰り返す場合，以降の第一段階の回答の品質にも影響を与えると考えられる．そこで，ワーカが自己補正タスクを繰り返す実験により，ワーカ自身の回答精度に影響をもたらすかを検証する．

要約すると，この実験では自己補正タスクに取り組んだ現実のワーカのタスク結果について，次の３点を検討する．

表 5.1: 各実験の設定の比較

	実験 1A	実験 1B	実験 2
目的	自己補正の参考回答 の有無が重要か確かめる	自己補正の参考回答 の品質が重要か確かめる	自己補正による学習 の転移が見られるか 確かめる
比較する条件	参考回答あり と 参考回答なし	参考回答が正解 と 参考回答がランダム	自己補正タスク と 通常のタスク
学習タスク数	28	52	48
テストタスク	学習タスクと同じ データセットで作成	学習タスクと同じ データセットで作成	学習タスクと異なる データセットで作成
タスク	鳥の画像分類	絵画の分類	鳥の画像分類
タスクの難易度	簡単	難しい	簡単, 難しい
分析の対象	テストタスクの 正答率が 25%以上	回答既知タスクに正解	回答既知タスクに正解

1. 現実のクラウドソーシングタスクに自己補正を導入することで，ワーカから得られる回答の品質が改善されるか
2. 自己補正タスクによる回答品質の改善のために，参考回答として提示する回答の有無は重要な要因であるか
3. ワーカが自己補正タスクを繰り返すことにより，ワーカ自身の回答品質が改善されるか

実験方法

４択の画像分類課題を用いた実験を行った．以下に実験の内容を詳述する．

扱う課題　実験参加者は選択式の画像分類タスクを行なった．選択肢は 4 種類で構成され，選択肢は全タスクを通して共通とした．タスクでは鳥類の画像のデータセットである Caltech-UCSD Birds 200[63] を用いた．このデータセットには 200 種類の鳥について複数の画像が含まれている．データセットには鳥の種類毎に複数の画像が含まれているため，タスクの難易度を調節するために，容姿のよく似た種類の鳥を 4 種類選択した．提示される画像はワーカ間で共通であるが，出題する順番はワーカ毎に並び替えた．

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．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

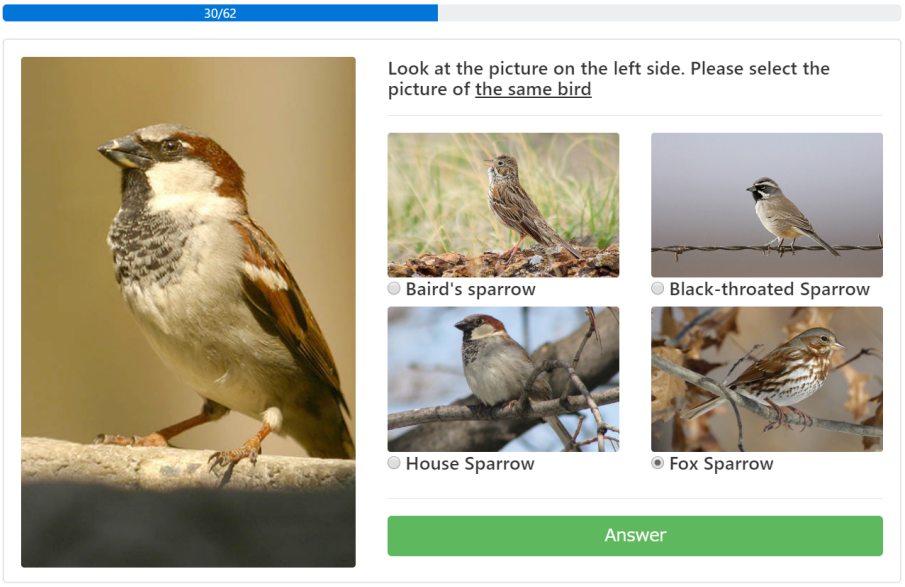


図 5.5: 実験 1A で用いるテストタスクの一例

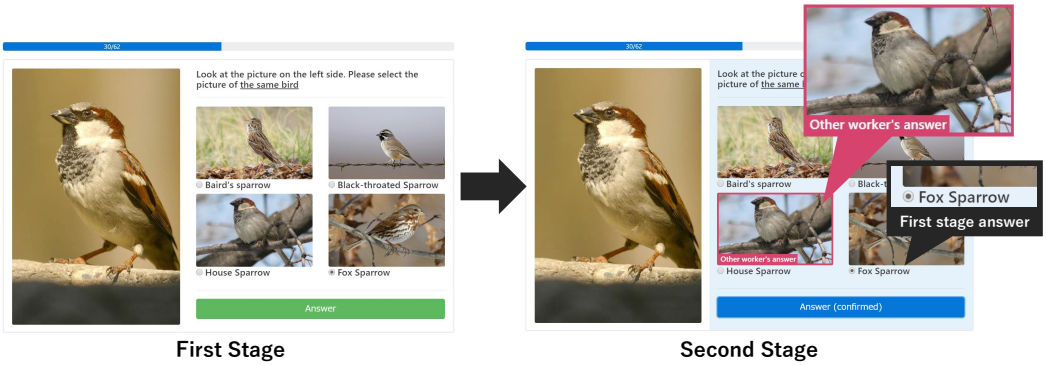


図 5.6: 実験 1A で用いる自己補正タスクの一例（研究業績 1-2 より転載）

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．

テストタスク　テストタスクの例を図 5.5 に示す．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．

自己補正タスク　自己補正タスクの例を図 5.6 に示す．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

表 5.2: 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	28
3	Mid テスト	テスト	12
4	学習 2	自己補正	28
5	Post テスト	テスト	12

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チェックボックス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．ステージ２で提示する参考回答には同じタスクに回答した別のワーカの回答を用いる．

比較する条件　実験では，自己補正のステージ２で提示する参考回答の重要性を明らかにするために，参考回答を提示する条件（以降では trusted 条件と呼ぶ）と参考回答を提示しない条件（以降では self 条件と呼ぶ）についてタスクの正答率や反応時間などを比較する．以下に，それぞれの条件について詳述する．

self 条件　この条件では，自己補正のステージ２において，参考回答を提示しない．ワーカはステージ２において，ステージ１での自身の回答のみが確認できる．この条件では，ステージ２における参考回答のハイライトは行われない．

trusted 条件　この条件では，自己補正のステージ２において，参考回答を提示する．ワーカはステージ２において，ステージ１での自身の回答に加えて，同じ質問に回答した他者の回答が確認できる．ステージ２での参考回答は赤い枠で示される．提示する参考回答には，self 条件に参加したワーカについて，全タスクの正答率を算出し，上位 20%のワーカのタスク結果を用いる．正答率が上位 20%に該当したワーカのうち，ランダムに選ばれたワーカの回答が自己補正のステージ２にて提示される．この条件における参考回答はあくまで他者の回答として提示され，提示された参考回答の信頼性に関してワーカには事前に知ることが出来ないものとする．

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 5.2 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

表 5.3: 実験 1A における Pre テストの成績及び総作業時間（秒）

			Pre-test Accuracy			Overall Working Time		
Condition	Filter	N	Median	Mean	Std	Median	Mean	Std
self	None	98	0.833	0.826	0.147	487.07	538.19	193.67
	Under 25%	84	0.833	0.83	0.136	517.44	553.74	202.05
trusted	None	98	0.833	0.816	0.131	526.01	549.60	175.88
	Under 25%	86	0.833	0.824	0.134	530.47	563.17	179.90

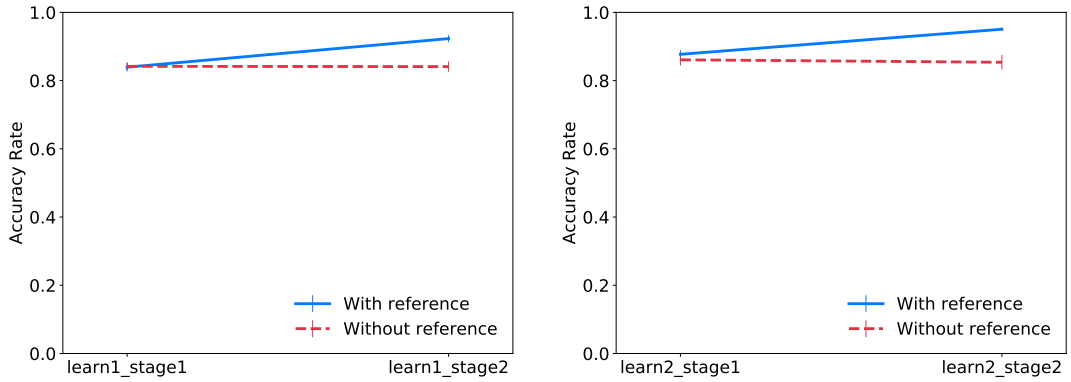


図 5.7: 参考回答の各条件における，自己補正の各ステージの正答率（左：学習フェーズ 1，右：学習フェーズ 2）

の効果を測定する．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割り当てた．

実験結果

実験参加者 196 名からのデータが得られた．得られたデータのうち 26 名のデータを除外してデータの分析を行った．実験では，ワーカは複数のタスクに連続で取り組むため，途中から無作為に回答を選ぶようなワーカが見られた．そこで，mid テスト及び post テストの平均正答率が 25%を下回るワーカについては分析の対象から除外した．表 5.3 に条件毎の実験参加者数と除外した人数，pre テストの正答率の平均値を示す．各実験参加者は約 9 分で一連のタスクを完了した．

自己補正の短期的効果　参考回答の条件毎の，学習フェーズ 1 における自己補正の各ステージの正答率を図 5.7 左に示す．同様に，学習フェーズ 2 における自己補正の各ステージの正答率を図 5.7 右に示す．これらのグラフの横軸は自己補正の各ステージで，縦軸は正答率を表している．

参考回答および学習フェーズ 1 での自己補正タスクのステージの違いによってタスクの正答率に差があるかを検証するために，独立変数を参考回答とステージ，従

属変数をタスクの正答率とする混合計画の２要因の分散分析を行った．その結果，参考回答要因の主効果およびステージ要因の主効果，そして交互作用が有意であった ($F(1, 168) = 6.578$, $p < .05$; $F(1, 168) = 33.74$, $p < .001$; $F(1, 168) = 34.31$, $p < .001$)．まず，ステージ期要因の各水準における参考回答要因の単純主効果の検定を行ったところ，ステージ 1 水準においては単純主効果が認められなかったが，ステージ 2 水準では有意な単純主効果が認められた ($F(1, 168) = .023$, n.s.; $F(1, 168) = 22.37$, .001)．次に，参考回答要因の各水準におけるステージ要因の単純主効果の検定を行ったところ，trusted 水準では有意な単純主効果が認められた ($F(1, 85) = 39.63$, $p < .001$) が self 水準では単純主効果が認められなかった ($F(1, 83) = 0.026$, n.s.)．

参考回答および学習フェーズ 2 での自己補正タスクのステージの違いによってタスクの正答率に差があるかを検証するために，独立変数を参考回答とステージ，従属変数をタスクの正答率とする混合計画の２要因の分散分析を行った．その結果，参考回答要因の主効果およびステージ要因の主効果，そして交互作用が有意であった ($F(1, 168) = 10.9$, $p < .01$; $F(1, 168) = 33.19$, $p < .001$; $F(1, 168) = 47.86$, $p < .001$)．まず，ステージ期要因の各水準における参考回答要因の単純主効果の検定を行ったところ，ステージ 1 水準においては単純主効果が認められなかったが，ステージ 2 水準では有意な単純主効果が認められた ($F(1, 168) = .761$, n.s.; $F(1, 168) = 30.04$, .001)．次に，参考回答要因の各水準におけるステージ要因の単純主効果の検定を行ったところ，trusted 水準では有意な単純主効果が認められた ($F(1, 85) = 51.89$, $p < .001$) が self 水準では単純主効果が認められなかった ($F(1, 83) = 1.725$, n.s.)．

参考回答および各学習フェーズでの自己補正タスクのステージ１においてタスクの正答率に差があるかを検証するために，独立変数を参考回答と学習フェーズ，従属変数をタスクの正答率とする混合計画の２要因の分散分析を行った．その結果，学習フェーズ要因の主効果に有意差が認められ ($F(1, 168) = 16.239$, $p < .001$)，参考回答要因と交互作用には有意差が認められなかった ($F(1, 168) = .175$, n.s.; $F(1, 168) = 1.731$, n.s)．

自己補正の長期的効果　参考回答の条件毎の，各テスト時期の正答率を図 5.8 に示す．図 5.8 の横軸はテスト時期で，縦軸は正答率を表している．

参考回答およびテスト時期の違いによってタスクの平均正答率に差があるかを検証するために，独立変数を参考回答とテスト時期，従属変数をタスクの正答率とする混合計画の 2 要因の分散分析を行った．その結果，テスト時期要因の主効果および交互作用が有意であった ($F(2, 336) = 8.831$, $p < .001$; $F(2, 336) = 3.5$, $p < .05$;) が，参考回答要因の主効果は有意ではなかった ($F(1, 168) = 0.635$, n.s.)．まず，テスト時期要因の各水準における参考回答要因の単純主効果の検定を行ったところ，post テストにおいて有意な単純主効果が認められたが，pre テストおよび mid テス

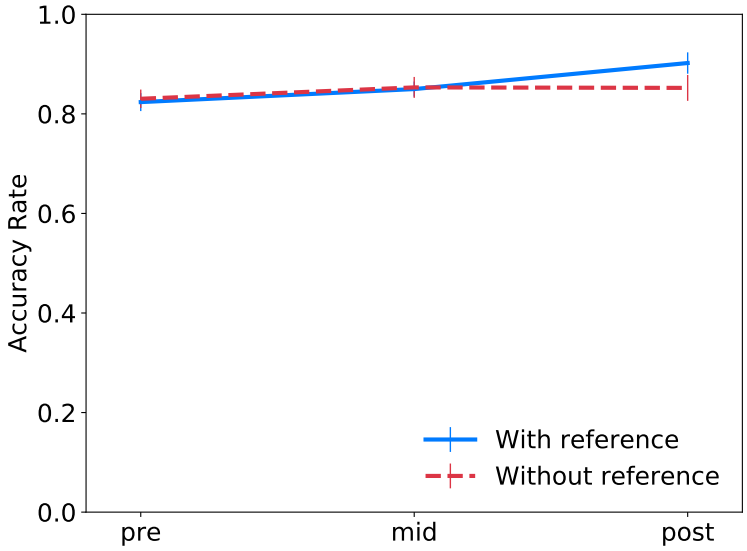


図 5.8: 参考回答の各条件における，テスト時期ごとの正答率

トでは単純主効果が認められなかった (pre: F(1, 168) = .105, n.s.; mid: F(1, 168) = .027, n.s.; post: F(1, 168) = 4.475, p <.05)．次に，参考回答要因の各水準におけるテスト時期要因の単純主効果の検定を行ったところ，trusted 水準では有意な単純主効果が認められた (F(2, 170) = 11.82, p <.001) が self 水準では単純主効果が認められなかった (F(2, 166) = 1.08, n.s.)．trusted 水準における各テスト時期に対してボンフェロー二の方法による多重比較を行ったところ，post 水準と pre 水準の間，および post 水準と mid 水準の間に有意差が認められ，mid 水準と pre 水準の間には有意差が認められなかった．

反応時間（ステージ要因）　参考回答の条件毎の，学習フェーズ 1 における自己補正の各ステージの反応時間を図 5.9 に示す．同様に，学習フェーズ 2 における自己補正の各ステージの反応時間を図 5.10 に示す．図 5.9 および 5.10 の横軸は自己補正の各ステージで，縦軸は反応時間を表している．

参考回答および学習フェーズ 1 での自己補正タスクのステージの違いによってタスク回答での反応時間に差があるかを検証するために，独立変数を参考回答とステージ，従属変数をタスクの反応時間とする混合計画の 2 要因の分散分析を行った．その結果，参考回答要因の主効果およびステージ要因の主効果，そして交互作用が有意であった (F(1, 168) = 5.17, p <.05; F(1, 168) = 5814.9, p <.001; F(1, 168) = 15.3, p <.001)．まず，ステージ期要因の各水準における参考回答要因の単純主効果の検定を行ったところ，ステージ 1 水準においては単純主効果が認められなかったが，ステージ 2 水準では有意な単純主効果が認められた (F(1, 168) = .114, n.s.;

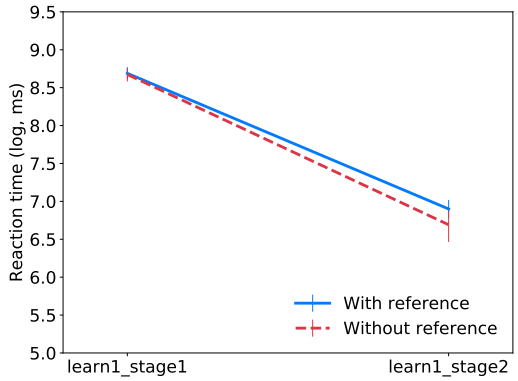


図 5.9: 参考回答の各条件における，自己補正の各ステージの反応時間（学習フェーズ 1）

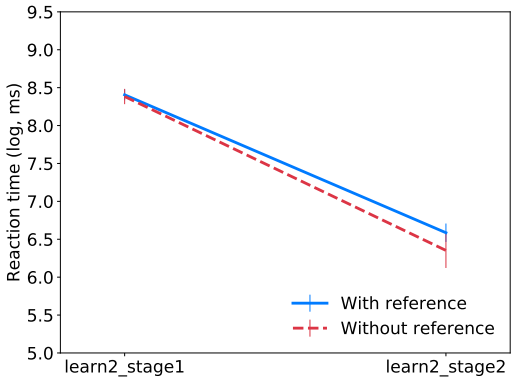


図 5.10: 参考回答の各条件における，自己補正の各ステージの反応時間（学習フェーズ 2）

F(1, 168) = 10.71, .001). 次に，参考回答要因の各水準におけるステージ要因の単純主効果の検定を行ったところ，trusted 水準と self 水準のそれぞれに有意な単純主効果が認められた (F(1, 85) = 3604, p <.001, F(1, 85) = 2502, p <.001).

参考回答および学習フェーズ 2 での自己補正タスクのステージの違いによってタスク回答での反応時間に差があるかを検証するために，独立変数を参考回答とステージ，従属変数をタスクの反応時間とする混合計画の 2 要因の分散分析を行った．その結果，参考回答要因の主効果およびステージ要因の主効果，そして交互作用が有意であった (F(1, 168) = 6.581, p <.05; F(1, 168) = 6227.94, p <.001; F(1, 168) = 18.76, p <.001). まず，ステージ期要因の各水準における参考回答要因の単純主効果の検定を行ったところ，ステージ 1 水準においては単純主効果が認められなかったが，ステージ 2 水準では有意な単純主効果が認められた (F(1, 168) = .246, n.s.; F(1, 168) = 13.17, .001). 次に，参考回答要因の各水準におけるステージ要因の単純主効果の検定を行ったところ，trusted 水準と self 水準のそれぞれに有意な単純主効果が認められた (F(1, 85) = 3285, p <.001, F(1, 85) = 2993, p <.001).

参考回答および各学習フェーズでの自己補正タスクのステージ 1 における反応時間に差があるかを検証するために，独立変数を参考回答と学習フェーズ，従属変数をタスク回答の反応時間とする混合計画の 2 要因の分散分析を行った．その結果，学習フェーズ要因の主効果に有意差が認められ (F(1, 168) = 435.543, p <.001) ，参考回答要因と交互作用には有意差が認められなかった (F(1, 168) = .192, n.s.; F(1, 168) = .068, n.s).

反応時間（テスト時期要因）　参考回答の条件毎の，各テスト時期の反応時間を図 5.11 に示す．図 5.11 の横軸はテスト時期で，縦軸は反応時間を表している．

参考回答およびテスト時期の違いによってタスク回答の平均反応時間に差がある

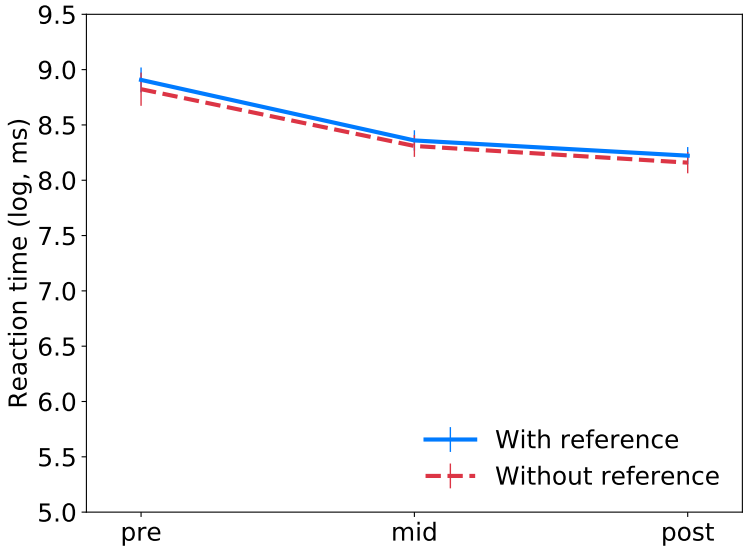


図 5.11: 参考回答の各条件における，テスト時期ごとの反応時間

かを検証するために，独立変数を参考回答とテスト時期，従属変数をタスク回答の反応時間とする混合計画の 2 要因の分散分析を行った．その結果，テスト時期要因の主効果が有意であった ($F(2, 336) = 614.783$, $p < .001$) が，参考回答要因の主効果および交互作用は有意ではなかった ($F(1, 168) = 2.255$, n.s.; $F(1, 168) = 0.356$, n.s.)．各テスト時期に対してボンフェロー二の方法による多重比較を行ったところ，post 水準と pre 水準の間および post 水準と mid 水準の間，mid 水準と pre 水準の間のすべてに有意差が認められた．

反応時間と正答率の関係　自己補正の各ステージの正答率と反応時間の相関関係を分析した．trusted 条件において，ステージ 1 およびステージ 2 の正答率と反応時間の相関は見られなかった (stage 1: r=-0.067, stage 2: r=-0.047)．同様に self 条件においても相関は見られなかった (stage 1: r=0.085, stage 2: r=-0.050)．

さらに，テストフェーズの成績と反応時間の相関関係を調べた．trusted 条件において pre 及び post テストの成績と反応時間の相関は見られなかった (pre: r=0.151, post: r=-0.047)．同様に self 条件においても相関は見られなかった (pre: r=-0.189, post: r=-0.016)．

これらの結果から，タスクの正答率と反応時間の間の相関は観察されなかった．この実験では，平均正答率が高く，ワーカが瞬時に回答できるようなタスクを用いたことから，反応時間に対する実験条件などの影響は少なかったと考えられる．

ワーカの成長度合い　ワーカの成長度合いを，post テストの正答率から pre テストの正答率を引いた値として考える．参考回答要因の各条件における，ワーカの成長度合

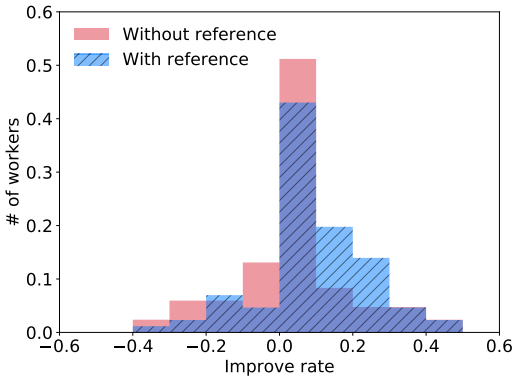


図 5.12: 参考回答の条件毎の，ワーカの成長度合いの分布（業績 2-2 より転載）

いの分布を図 5.12 に示す．横軸は成長度合いを，縦軸は該当するワーカの割り合いを示す．

参考回答要因の各水準における成長度合いの分布に差があるかを確認するために，マンホイットニーの U 検定を行ったところ有意差が認められた ($p < .05$)．

考察

自己補正の短期的効果　学習フェーズ 1 での結果について考察する．自己補正のステージ要因の各水準における参考回答要因の単純主効果の検定では，ステージ 1 では有意差が認められず，ステージ 2 では有意差が認められた．そして，参考回答要因の各水準における，自己補正のステージ要因の単純主効果の検定では，trusted 条件では有意差が認められたが，self 条件では有意差が認められなかった．このことから，trusted 水準で用いた様な性質を持つ参考回答を提示することで，自己補正のステージ 2 での正答率が改善されたと言える．trusted 水準における各ステージ水準での正答率は，Shah らの主張であるタスクに自己補正を導入することによる正答率の改善を支持するものである．

自己補正タスクによる正答率の改善は，self 条件では見られず，trusted 条件でのみ見られたことから，参考回答の有無は重要な要因の 1 つであると考えられ，より信頼性の高い参考回答を提示することが求められると言える．ただし，一部のワーカは自己補正のステージ 2 において，常に参考回答として提示された回答へと変更するような振る舞いをすると考えられる．加えて，ワーカが参考回答を活用できるかどうかは，タスク自体の難易度や，ワーカがタスクで問われている内容を理解しているかなどに依存すると考えられる．

学習フェーズ 2 においても，学習フェーズ 1 と同様の傾向が見られた．学習フェーズ 2 における傾向の考察については，学習フェーズ 1 と同様の説明が可能であると考えられるため，説明は省略する．

学習フェーズ 1 および学習フェーズ 2 での自己補正タスクのステージ 1 の正答率に主効果が認められた．しかし，参考回答要因の主効果および交互作用の有意差が認められなかった．このことから，学習フェーズ 2 の自己補正タスクのステージ 1 での正答率の改善はタスクへの回答の繰り返しによるものであると考えられる．

自己補正の長期的効果　テスト時期要因の各水準における参考回答要因の単純主効果の検定では，pre 水準および mid 水準には有意差が認められず，post 水準では有意差が認められた．そして，参考回答要因の各水準におけるテスト時期要因の単純主効果の検定では，trusted 水準では有意差が認められ，self 水準では有意差が認められなかった．つまり，trusted 水準で用いた様な性質を持つ参考回答を提示することで，pre 水準の正答率を比べて post 水準の成績が上回ったと言える．このことから，ワーカが信頼できる参考回答を提示する自己補正タスクに連続で取り組むことで，ワーカ自身の正答率が改善することが示唆された．視覚的な作業を行う能力は，繰り返すことにより，すなわち知覚的な学習によりその速度や精度が改善されることが知られている [59]．知覚学習は意図せず無自覚に生じるものであり [60]，複数の研究が，視覚的な分類課題における知覚学習を報告している [61]．このことから，自己補正の長期的効果は無自覚な知覚学習として説明できると考えられる．

trusted 水準における各テスト時期の成績の多重比較では，mid 水準と pre 水準の間には有意差が認められず，post 水準と mid 水準の間および post 水準と pre 水準の間に有意差が認められた．このことから，ワーカの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考えれる．ただし，この傾向は今回の実験の設定の範囲内で主張できることであり，自己補正を繰り返す回数やタスクで扱う課題などによって成長の度合いが左右されることが予想される．

反応時間（ステージ要因）　学習フェーズ 1 における，自己補正のステージ要因の各水準における参考回答要因の単純主効果の検定では，ステージ 1 では有意差が認められず，ステージ 2 では有意差が認められた．そして参考回答要因の各水準におけるステージ要因の単純主効果の検定では，trusted 水準と self 水準の両者に有意差が認められた．このことから，自己補正タスクで提示する参考回答の有無にかかわらず，ステージ 2 の反応時間はステージ 1 よりも短くなることが示唆された．更に，ステージ 2 における反応時間は self 水準と比較して trusted 水準のほうが有意に長いことが示唆された．ワーカは参考回答を提示された際に，自身の回答と提示された回答のどちらがより正しいかを判断するため，参考回答を提示する条件のほうが反応時間が長くなったと考えられる．

学習フェーズ 2 においても，学習フェーズ 1 と同様の傾向が見られた．学習フェーズ 1 と同様の議論ができると考えられるため，説明は省略する．

学習フェーズ1および学習フェーズ2での自己補正タスクのステージ1の反応時間率に主効果が認められた．しかし，参考回答要因の主効果および交互作用の有意差が認められなかった．このことから，学習フェーズ2の自己補正タスクのステージ1での反応時間の短縮はタスクへの回答の繰り返しによるものであると考えられる．

反応時間（テスト時期要因）　テスト時期要因の主効果が認められた．テスト時期要因の各水準の多重比較では，全ての水準間に有意差が認められた．ただし，参考回答要因の主効果および交互作用についての有意差は認められなかった．このことから，テスト時期要因の反応時間の短縮は，作業の繰り返しによるものであると考えられる．

ワーカの成長度合いの分布　テスト時期における post の成績から pre の成績を引いた値をワーカの成長度合いと考える．各ワーカの成長度合いについてのヒストグラムを図 5.12 に示す．参考回答が trusted の条件では，成長度合いが 0.2 から 0.4 に相当するワーカの数が，self の条件よりも多いことが分かる．このことから，trusted の参考回答を提示したことにより，一部のワーカについては回答品質の改善に繋がったと考えられる．

5.3.2　実験 1B（参考回答の品質の影響）

目的

実験 1A では，鳥の画像分類タスクに対して自己補正を適用する実験を行った．その結果，参考回答を提示することでタスク結果の改善が見られた．さらに，自己補正を繰り返すことで，ワーカ自身の回答品質も改善されることが示唆された．しかし，参考回答要因の条件に関わらず，pre テストの平均正答率が高くなるような課題であったため，ワーカ自身の正答率の改善の幅が小さかった．そこで，この実験では，タスクの平均正答率が実験 1A よりも低くなるような課題を扱い，画像分類課題を行った．

実験 1A の結果から，自己補正において参考回答はタスク結果の改善およびワーカの学習において重要な要素であることが示唆された．しかし，参考回答の品質が，タスク結果の改善およびワーカの学習に与える影響は不明である．そこでこの実験では，参考回答として常に正答を提示する Correct 水準と常にランダムな回答を提示する Random 水準を比較することで，参考回答の品質がもたらす影響を明らかにする．

加えて，実験 1A におけるワーカの学習は pre テスト及び mid テストでは見られず，mid テストおよび post テストで見られたことから，ある程度の自己補正の繰り返しが必要であると考えられる．そのためこの実験では，各学習フェーズで割り当

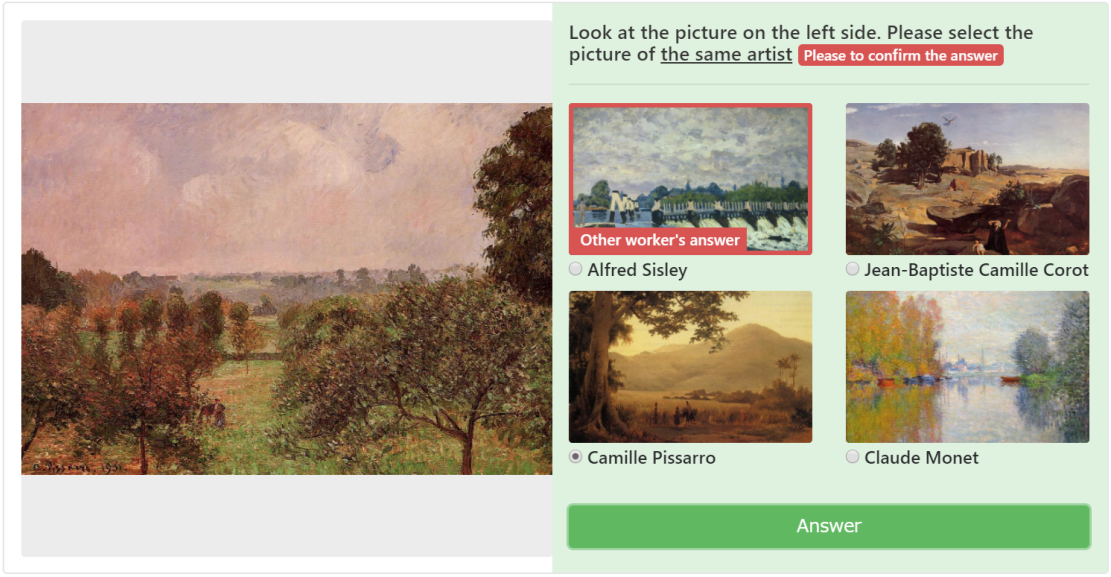


図 5.13: 実験 1B で用いる自己補正タスクの一例（画像はステージ 2 の状態のみ）.（研究業績 1-2 より転載）

てるタスク数を増やすことでワーカの学習の度合いがどの様に変化するかを明らかにする．実験 1A のワーカの成長度合いの分析から，post テストの成績が顕著に向上するような一部のワーカが存在すると考えられる．そこで，自己補正タスクでの回答の変更のパターンを分析することで，どのような性質のワーカに対して自己補正タスクが有効であったかを分析する．

実験方法

以下に実験内容について詳述する．

実験環境　全作業を完了したワーカであるかを識別するために用いるキーワードとトークンについて，すべてのワーカに共通なキーワードについては新たに生成した文字列を使用した．

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．実験では次の 4 名の画家の作品の画像データを用いた：(1) Alfred Sisley, (2) Jean-Baptiste Camille Corot, (3) Camille Pissarro, (4) Claude Monet.

タスク　実験 1A と同様にテストタスクと自己補正タスクを組み合わせて実験を行う．図 5.13 に自己補正タスクのステージ 2 の一例を示す．

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

表 5.4: 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 + 2 回答既知タスク
3	Mid テスト	テスト	12
4	学習 2	自己補正	52 + 2 回答既知タスク
5	Post テスト	テスト	12

比較する条件　実験では，自己補正のステージ 2 で提示する参考回答の品質の重要性を明らかにするために，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は反応時間を比較した．実験 1A と同様に，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．

ワーカのフィルタリング　実験１ A では，ランダムな回答を行うワーカを分析の対象から除外するために，mid テストおよび post テストの成績に基づいてフィルタリングを行った．この実験では，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．

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 5.4 に示す．フェーズの構成自体は実験 1A と同様であるが，学習フェーズにおけるタスク数が異なる．

回答変更率　この実験の分析では，自己補正タスクに取り組むワーカの回答の変更の仕方に着目する．ある実験参加者の自己補正タスクへの回答の集合を T とする．各自己補正タスクの要素 $t \in T$ はステージ 1 での回答 t_{stage1} とステージ 2 の回答 t_{stage2} を持つ．ここで，ある t において $stage1$ と $stage2$ の値が異なる場合，ワーカはそのタスクで解答を変更したことを意味する．一連の自己補正タスクにおけるあるワーカの回答変更率は式 5.1 により求めることができる．

$$\text{回答変更率} = \frac{|\{t | t_{stage1} \neq t_{stage2}, t \in T\}|}{|T|} \tag{5.1}$$

表 5.5: 実験 1B における Pre テストの成績および総作業時間（秒）

			Pre-test Accuracy			Overall Working Time		
Condition	Filter	N	Median	Mean	Std	Median	Mean	Std
random	None	105	0.333	0.354	0.152	823.79	902.76	482.83
	Gold	99	0.333	0.357	0.153	832.65	923.78	473.50
correct	None	86	0.333	0.356	0.15	876.06	898.56	379.89
	Gold	82	0.333	0.363	0.153	885.90	914.27	373.76

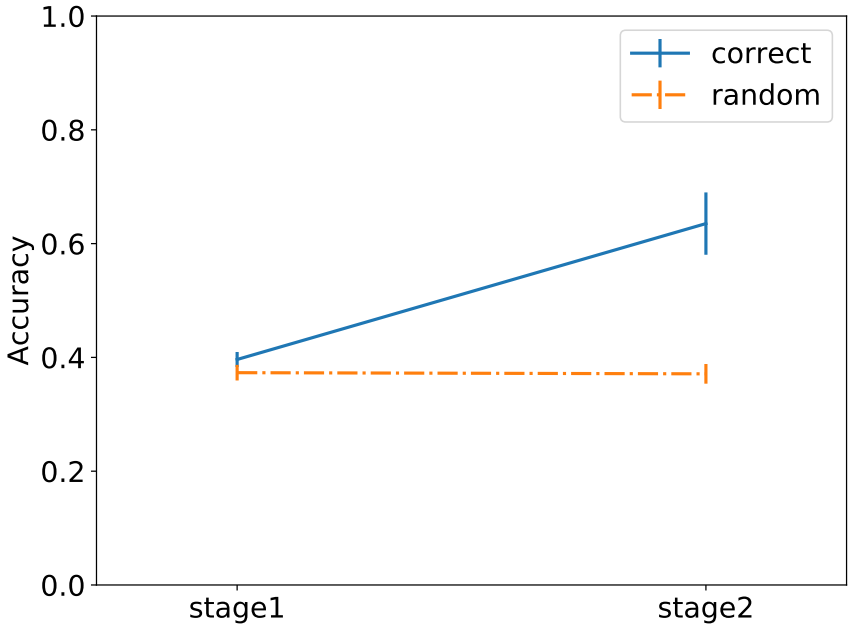


図 5.14: 参考回答の各条件における，自己補正の各ステージの正答率（研究業績 1-2 より転載）

実験結果

実験参加者 191 名からのデータが得られた．得られたデータのうち，10 名のデータを除外してデータの分析を行った．除外したのは，学習フェーズ中に出題した，選択肢と同様の画像が出題されるタスクについて，正答出来なかったワーカである．表 5.5 に，条件毎の実験参加者数と分析から除外した人数，各セクションの正答率の平均値を示す．各ワーカは一連のタスクを完了するのに約 15 分を費やした．

自己補正の短期的効果　参考回答の条件ごとの，全学習フェーズを通した自己補正の各ステージの正答率を図 5.14 に示す．

参考回答の条件及び自己補正のステージによりタスクの正答率に差があるかを明らかにするために，独立変数を参考回答の条件とステージ，従属変数をタスクの正答率とする混合計画の 2 要因の分散分析を行った．その結果，参考回答要因の主効果

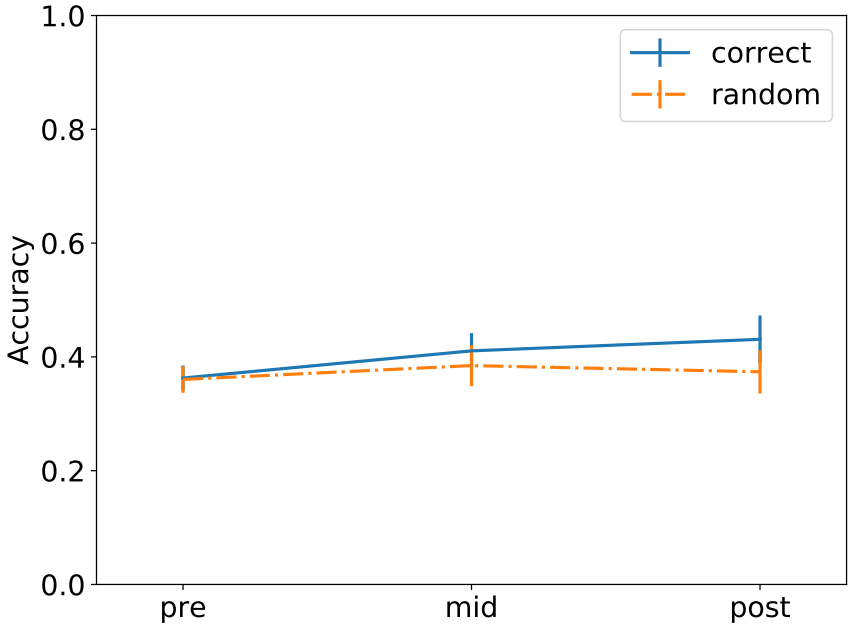


図 5.15: 参考回答の各条件における，テスト時期ごとの正答率（研究業績 1-2 より転載）

およびステージ要因の主効果，そして交互作用が有意であった ($F(1, 179) = 52.34$, $p < .001$; $F(1, 179) = 83.97$, $p < .001$; $F(1, 179) = 105.27$, $p < .001$).

まず，参考回答要因の各水準におけるステージ要因の単純主効果の検定を行ったところ，correct 水準では有意な単純主効果が認められた ($F(1, 81) = 91.05$, $p < .001$) が random 水準では単純主効果が認められなかった ($F(1, 98) = 0.15$, n.s.). 次に，ステージ期要因の各水準における参考回答要因の単純主効果の検定を行ったところ，ステージ 1 水準においては単純主効果が認められなかったが，ステージ 2 水準では有意な単純主効果が認められた ($F(1, 179) = 1.79$, n.s.; $F(1, 179) = 91.32$, $p < .001$).

自己補正の長期的効果　参考回答の条件毎の，各テスト時期の正答率を図 5.15 に示す．図の横軸はテスト時期で，縦軸は正答率を表している．

参考回答およびテスト時期の違いによってタスクの平均正答率に差があるかを検証するために，独立変数を参考回答とテスト時期，従属変数をタスクの正答率とする混合計画の 2 要因の分散分析を行った．その結果，テスト時期要因の主効果が有意であった ($F(2, 358) = 3.89$, $p < .05$) が，参考回答要因の主効果および交互作用は有意ではなかった ($F(1, 179) = 2.016$, n.s.; $F(2, 358) = 1.61$, n.s.).

交互作用が見られなかったので，各条件における多重比較を行った．correct 条件におけるテスト時期要因に対してボンフェロー二の方法による多重比較を行ったところ，pre 水準と post 水準の間には有意差が認められたが ($p < 0.05$)，pre 水準およ

表 5.6: 実験 1B の回答変更率毎のワーカ数

		condition	
group	answer change rate	correct	random
1	0.0 ∼ 0.2	41	65
2	0.2 ∼ 0.4	21	14
3	0.4 ∼ 0.6	11	9
4	0.6 ∼ 0.8	9	9
	0.8 ∼ 1.0	0	2

び mid 水準の間と mid 水準および post 水準の間には有意差が認められなかった.

random 条件におけるテスト時期要因に対して同様に多重比較を行ったところ，全ての水準間に有意差が認められなかった.

回答変更率　実験 1B における自己補正の長期的効果は，実験 1A と比較して小さかった．ただし，タスクに意欲的に取り組むワーカとそうでないワーカでは学習効果の大きさに差が生じると想定される．つまり，タスクに意欲的に取り組んだワーカの方が学習効果が大きくなると考えられる.

ここで，ワーカのタスクへの意欲は自己補正タスクのステージ２において自身が決定したステージ１の回答を変更した割合に現れると仮定する．意欲的なワーカは，ステージ２で提示される参考回答を慎重に検討し，回答の変更の機会を有効活用できると考えられる．一方で，提示された参考回答を無視ステージ１の回答を維持し続けたり，常に参考回答へと変更するようなワーカは意欲が低いと見なすことができる.

そこで，実験結果を自己補正タスクにおける回答変更率でいくつかのグループに分け，グループ間での学習効果の違いを分析した.

表 5.6 に，各条件の回答変更率のグループごとのワーカの人数の分布を示す．以降の分析では回答変更率が 0.8 1.0 のグループを除く，４つのグループに着目する．これはこのグループに該当するワーカの人数が少ないためである.

図 5.16 に参考回答の各条件における，回答変更率のグループ毎のテストタスクへの成績を示す.

correct 条件のグループについて，独立変数を回答変更率のグループとテスト時期，従属変数をタスクの正答率とする混合計画の２要因の分散分析を行った．その結果，テスト時期要因の主効果が認められなかった ($F(2, 156) = 2.07$, n.s.) が，回答変更率の主効果および交互作用が認められた ($F(3, 78) = 5.88$, $p < .005$; $F(6, 156) = 2.26$, $p < .05$).

テスト時期要因の各水準における回答変更率要因の単純主効果の検定を行ったところ，グループ 2 の水準で単純主効果が認められた (回答変更率 0.2 - 0.4, $p < .005$)

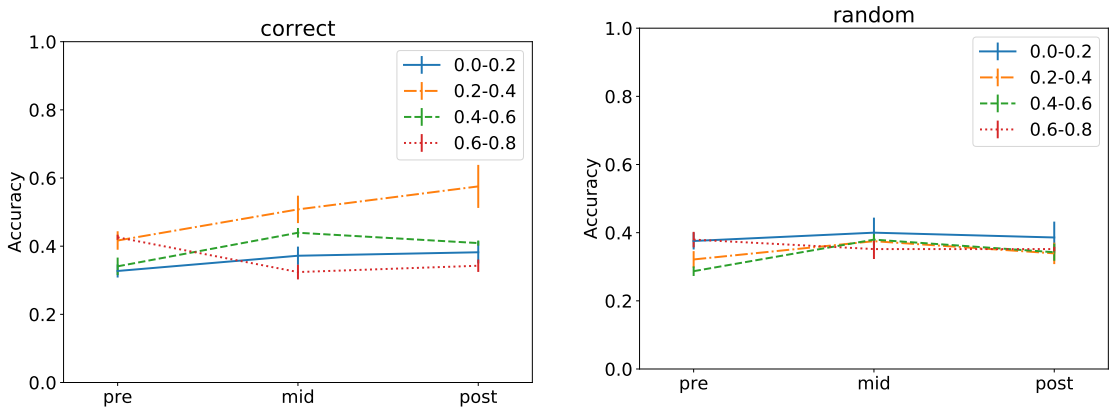


図 5.16: 回答変更率とテストタスクの正答率の関係（研究業績 1-2 より転載）

．多重比較を行ったところ，pre テストと post テストの間に有意差が認められた（$p < .01$）．

回答変更率要因の各水準におけるテスト時期要因の単純主効果の検定を行ったところ，mid テストと post テストの水準で単純主効果が認められた ($F(3, 78) = 4.00$, $p < .05$; $F(3, 78) = 5.70$, $p < .005$)．mid テストにおいて多重比較を行ったところ，グループ 1 とグループ 2 ($p < .05$)，およびグループ 2 とグループ 4 ($p < .05$) の間で有意差が認められた．post テストにおける多重比較においても，グループ 1 とグループ 2 ($p < .05$) およびグループ 2 とグループ 4 ($p < .05$) の間に有意差が認められた．

random 条件のグループについて，独立変数を回答変更率のグループとテスト時期，従属変数をタスクの正答率とする混合計画の 2 要因の分散分析を行ったところ，テスト時期と回答変更率のグループおよび交互作用が認められなかった p($F(2, 186) = 0.74$, n.s.; $F(3, 93) = 0.69$, n.s.;$F(6, 186) = 0.29$, n.s.).

反応時間と正答率の関係

自己補正の各ステージでの正答率と反応時間の相関関係を分析した．correct 条件では，ステージ 1 とステージ 2 でそれぞれ弱相関と中程度の相関が見られた (stage 1: r=0.331, stage 2: r=0.419)．ただし，random 条件ではステージ 1 では中程度の相関 (r=0.412) が見られるものの，ステージ 2 で相関が見られなかった (r=0.145).

さらに，各テスト時期における反応時間との相関関係を分析した．correct 条件では，pre テストでは相関が見られなかった (r=-0.052) が，post テストでは弱相関が見られた (r=0.383)．random 条件でも同様に，pre テストでは相関が見られなかった (r=0.061) が，post テストでは弱相関が見られた (r=0.321).

実験 1B では，実験 1A よりも難しいタスクを用いて実験を行った．その結果，実験

1A と比べて反応時間と正答率の間に弱い関係が観察された．実験 1B の特に correct 条件では，タスクの難易度が高いことから，ワーカのステージ１の回答と参考回答が一致しない場合の頻度が高くなる．真面目に作業に取り組むワーカはそれらを見比べるため，作業時間が増加するものと考えられる．

考察

自己補正の短期的効果　自己補正のステージ要因の各水準における参考回答要因の単純主効果の検定では，ステージ１では有意差が認められず，ステージ２では有意差が認められた．そして，参考回答要因の各水準における，ステージ要因の単純主効果の検定では，correct 水準では有意差が認められたが，random 水準では有意差が認められなかった．correct 水準における結果は，Shah らの主張であるタスクに自己補正を導入することによる正答率の改善および，実験 1A の結果を支持するものである．このことから，参考回答の品質は自己補正がもたらす効果を左右する要因であることが示唆された．実験 1B では参考回答としてランダムおよび正答を，実験 1A では現実のワーカから得た回答を提示する実験を行ったが，参考回答の品質と自己補正によるタスク結果の改善の関係についてはこれらの結果のみから説明することは出来ない．

correct 水準においては，ステージ２の正答率が 100%に近づくことも予想されたが，ステージ１の正答率である 40%から 60%への改善に留まった．そして，random 水準に場合においてはステージ２の正答率がチャンスレベルである 25%に近づくことが予想されたが，ステージ２の正答率はステージ１と同程度であった．このことから，大半のワーカは参考回答を見た上で，自身の回答を変更するかを判断していると考えられる．実験 1A および実験 1B では，参考回答を単に他者の回答であるとして提示したが，例えば信頼できるワーカの回答であるなどと説明を付け加えることにより，参考回答に変更する割り合いが変化する可能性がある．

自己補正の長期的効果　参考回答要因とテスト時期要因の分散分析では，テスト時期要因の主効果にのみ有意差が認められた．多重比較では，pre 水準および post 水準の間にのみ有意差が認められた．このことから，この実験では実験 1A で述べた様なワーカの学習の効果は見られなかった．この原因の１つとしては，タスクの難易度を高く設定したことにより，ワーカが各選択毎の特徴を捉えたり，提示された参考回答の信頼性を判断するのが難しくなっていることが挙げられる．

自己補正タスクを割り当てる回数をより増やすことにより，学習の機会を増やすことで，交互作用が認められる可能性もあるが，一方で作業を中断したり離脱するワーカの存在が懸念される．

回答変更率　自己補正タスクの繰り返しによるテストタスクの正答率の改善（つまり長期的効果）は，参考回答として正解を提示する条件における回答変更率が 0.2　0.4 のグループでのみ確認された．このグループのワーカの正答率は 0.42 から 0.58 へと向上した．この結果は，自己補正タスクが全てのワーカの能力を向上させるとは限らないが，少なくとも特定の範囲のワーカに対して効果的である可能性を示唆するものである．

加えて，自己補正タスクに真面目に取り組んだワーカの回答変更率を観察することで，将来的に能力の向上が見込まれるワーカを推定できる可能性がある．この方法に基づいてワーカの能力を向上させる枠組みを設計するためには，ワーカが継続してタスクに取り組むための動機付けや報酬設計を検討する必要があると考えられる．

さらに，潜在的に活躍が見込まれるワーカが属する回答変更率のグループは，ワーカに割り当てるタスクの性質や難易度によって異なる可能性がある．

5.4　自己補正による学習の転移

5.4.1　実験 2

目的

実験 2 では，自己補正タスクの繰り返しによる長期的効果が，学習時とは別のデータセットのタスクにも作用するかを明らかにする．実験 1A と実験 1B では，自己補正の短期的効果と長期的効果が，異なる２つのデータセットでの実験から観察された．これらの実験では，学習フェーズのタスクとテストフェーズのタスクが同一の４クラスのデータセットで構成されていた．

それに対して実験２では，学習フェーズとテストフェーズで異なる４クラスのデータセットを用いる．この実験により，学習フェーズで自己補正タスクを経験した人間ワーカが，別のデータセットでの作業において，学習前よりも正確に作業できるかを，学習フェーズで自己補正タスクを用いない条件と比較することで検証する．このような，ある課題における学習が，類似した別の課題の学習に影響を与えることは学習の転移 [64] と呼ばれる．

学習の転移の生じやすさは，学習者の状態のみならず学習課題と転移課題の類似性，学習環境といった様々な要因の影響を受けると考えられる [65]．これらの要因の組み合わせにより，学習の転移が生じやすい状況のことを近い転移，学習の転移が生じにくい状況を遠い転移として分類される．つまり，学習課題と転移課題として，同等又は類似した課題を用いることは近い転移として説明することができる．

そこで実験 2 では，学習課題と転移課題の組み合わせ方として，平均正答率が高いデータセットの組と，平均正答率が低いデータセットの組を用意して比較した．実

図 5.17: 実験 2 で用いる自己補正タスクの一例（画面はステージ 2 の状態）.（研究業績 1-2 より転載）

験 1A と実験 1B の実験結果から，平均正答率が高くなるようなデータセットの方が自己補正による長期的効果が生じやすいと考えられ．タスクの難易度は学習の転移にも影響することが予想される．自己補正により学習の近い転移を促すことは，人間ワーカが作業を通して得られた学習効果を，同系統の別の作業に波及されることが期待でき，これはクラウドソーシングプラットフォーム全体の処理能力向上に繋がる．

実験方法

タスク　実験 1A と同様にテストタスクと自己補正タスクを組み合わせて実験を行う．図 5.17 に自己補正タスクのステージ 2 の一例を示す．

データセット　異なる選択肢で構成されるタスクを組み合わせて用いるにあたり，タスクの候補を複数作成し，Amazon Mechanical Turk[5]を利用して実際に一部のタスクの作業を依頼した．集められたタスク結果から，各データセットの正答率の傾向を求め，傾向が類似しているデータセットのペアを選びだした．データセット 1 とデータセット 2 は，平均正答率が 50%付近となるようなデータセットである．データセット 3 とデータセット 4 は平均正答率が 90%付近となるようなデータセットである．実験では，データセット 1 と 2 をペアに，データセット 3 と 4 をペアにして，学習タスクとテストタスクにそれぞれを適用することで，合計 4 種類の課題を作成した．

[5]https://www.mturk.com/

表 5.7: 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	24
2	学習 1	自己補正	48 + 4 回答既知タスク
3	Post テスト	テスト	24

比較する条件　実験では，自己補正によりワーカの学習が別の課題でも有効であるかを明らかにするために，自己補正のステージ２で正答を提示する SC 条件と，自己補正ではなくテストタスクと同等の NSC 条件について，タスク結果の正答率を比較した．

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 5.7 に示す．フェーズの要素自体は実験 1A と同様であるが，実験 1A では学習フェーズは２回であるが，この実験では１回である．更にテストフェーズで割り当てるタスク数が実験 1A とは異なる．

ワーカのフィルタ　実験 1B と同様の方法を用いた．学習フェーズに，選択肢に表示される画像が出題されるタスクを 4 回提示し，それら全てに回答できたワーカを分析の対象とした．

実験結果

実験参加者から 392 名のデータが得られた．得られたデータのうち，15 名のデータを除外して実験結果の分析を行った．除外したデータは，学習フェーズ中に出題される選択肢と同様の画像が出題されるタスクに正解することができなかったワーカの結果である．表 5.8 に，条件ごとの実験参加者数と除外した人数と pre テストの成績を示す．

以降の分析では，４種類のデータセットから得られた実験結果を扱う．多重検定の問題に対処するために，検定で用いる有意水準をボンフェローニ法により補正する．有意水準を 0.05 とした時の補正後の有意水準は 0.0125 である．

実験結果におけるワーカの識別子を確認したところ，実験 2 の参加者のうち 66 名が実験 1A にも参加したワーカであった．ただし，実験 2 は実験 1A の実施の 18ヶ月後に行われたものである．２つの実験間には十分な時間差があることから，実験 1A での学習効果は実験２に影響しないものと考える．

テスト要因の分析　各データセットにおける参考回答の条件毎の，各テスト時期の正答率を図 5.18 に示す．これらの図の横軸はテスト時期を，縦軸は正答率を表している．

表 5.8: 実験 2 の Pre テストの成績および総作業時間（秒）

				Pre-test Accuracy			Overall Working Time		
Dataset Pair	Condition	Filter	N	Median	Mean	Std	Median	Mean	Std
1, 2	NSC	None	43	0.375	0.381	0.128	693.33	733.11	364.89
(learning, test)		Gold	40	0.375	0.372	0.124	690.11	715.81	339.92
	SC	None	46	0.333	0.348	0.14	665.79	733.05	370.57
		Gold	42	0.417	0.354	0.145	674.32	751.56	358.67
2, 1	NSC	None	59	0.375	0.386	0.129	579.96	653.04	315.65
		Gold	56	0.375	0.395	0.125	588.99	658.82	304.44
	SC	None	44	0.417	0.447	0.106	710.77	796.44	339.48
		Gold	41	0.417	0.454	0.105	743.12	811.40	346.99
3, 4	NSC	None	49	0.75	0.736	0.152	445.94	482.46	173.15
		Gold	48	0.75	0.744	0.144	456.19	487.30	171.60
	SC	None	49	0.792	0.758	0.166	501.64	547.43	169.19
		Gold	48	0.792	0.767	0.153	514.98	552.27	167.51
4, 3	NSC	None	51	0.917	0.902	0.106	440.74	502.14	217.93
		Gold	51	0.917	0.902	0.106	440.74	502.14	217.93
	SC	None	51	0.917	0.895	0.091	581.45	658.53	288.87
		Gold	51	0.917	0.895	0.091	581.45	658.53	288.87

データセット 1 における，参考回答およびテスト時期によってタスクの正答率に差があるかを検証するために，独立変数を参考回答とテスト時期，従属変数をタスクの正答率とする混合計画の 2 要因の分散分析を行った．その結果，参考回答要因の主効果およびテスト時期要因の主効果，そして交互作用のすべてに有意差が認められなかった ($F(1, 63) = .214$, n.s.; $F(1, 63) = .162$, n.s.; $F(1, 63) = .049$, n.s.).

データセット 2 における，参考回答およびテスト時期によってタスクの正答率に差があるかを検証するために，独立変数を参考回答とテスト時期，従属変数をタスクの正答率とする混合計画の 2 要因の分散分析を行った．その結果，参考回答要因の主効果およびテスト時期要因の主効果，そして交互作用のすべてに有意差が認められなかった ($F(1, 68) = .947$, n.s.; $F(1, 68) = 1.082$, n.s.; $F(1, 68) = 2.630$, n.s.).

データセット 3 における，参考回答およびテスト時期によってタスクの正答率に差があるかを検証するために，独立変数を参考回答とテスト時期，従属変数をタスクの正答率とする混合計画の 2 要因の分散分析を行った．その結果，参考回答要因の主効果およびテスト時期要因の主効果，そして交互作用のすべてに有意差が認められなかった ($F(1, 91) = 2.571$, n.s.; $F(1, 91) = 4.838$, n.s.; $F(1, 91) = .019$, n.s.).

データセット 4 における，参考回答およびテスト時期によってタスクの正答率に差があるかを検証するために，独立変数を参考回答とテスト時期，従属変数をタスクの正答率とする混合計画の 2 要因の分散分析を行った．その結果，参考回答要因の主効果およびテスト時期要因の主効果，そして交互作用のすべてに有意差が認め

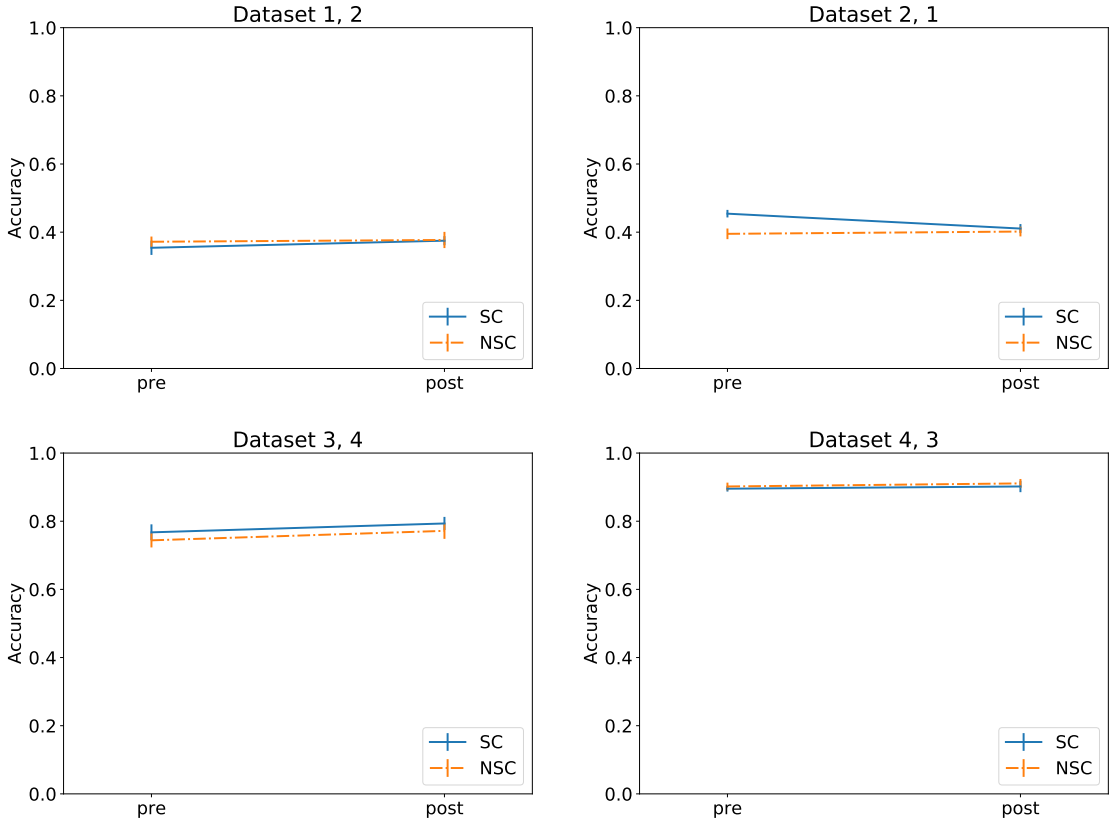


図 5.18: 各データセットにおける，条件毎の各テスト時期の正答率（研究業績 1-2 より転載）

られなかった ($F(1, 77) = .09$, n.s.; $F(1, 77) = 3.327$, n.s.; $F(1, 77) = .165$, n.s.).

考察

学習フェーズとテストフェーズで出題するタスクのデータセットを変えた実験では，自己補正の短期的および長期的効果は見られなかった．このことから，少なくともこれらの実験においては自己補正による学習の転移は見られなかった．この実験を発展させて自己補正による学習の転移を促すためには，データセットの選び方や学習タスク数を工夫する必要があると考えられる．

実験ではデータセットごとの平均正答率に基づいて，学習課題と転移課題の組み合わせを決定した．しかし，データセット間で，平均正答率の分布が類似していることが転移学習のしやすさと関係があるとは限らない．そのため，転移学習を促すためには，データセットごとに正確に作業をするために必要なドメイン知識などを考慮してデータセットを選ぶ必要があると考えられる．

5.5 結論

5.5.1 総合考察

クラウドソーシングにおいて，ワーカから得られる成果物の品質を管理することは重要な研究課題の１つである．これまでに多くの研究がこの課題に取り組んでおり，本研究ではその１つである Shah らが提案した自己補正に着目した．自己補正はタスク結果の品質を改善することを目的とした手法であり，１つのタスクに２度の回答の機会を与えることで，ワーカ自身が自分の誤答を補正できるのが特徴である．自己補正は，多数決をはじめとするタスク結果の集約手法や，優れたワーカの検出やタスク割り当て手法と組み合わせることが容易であることから，多くの場面で活用できる可能性がある．さらに，ワーカ自身の自らの誤答を気づかせる機会を与えることが，ワーカのその後の回答品質の改善にも効果があると期待できる．しかし自己補正が提案された論文では，シミュレーションによる評価のみが行われており，現実のクラウドソーシングでのワーカにもシミュレーションと同様の効果が得られるかは不明であった．

そこで本研究では，現実のクラウドソーシングワーカが自己補正を適用したタスクに取り組む実験により，自己補正の効果について検討した．実験 1A では自己補正の短期的な効果と長期的な効果を鳥の課題で検討した．実験 1B では，実験 1A で見られた効果が，別の難しいタスクにおいても同様の傾向であるかを，画家分類課題を用いて検討した．実験 2 では，特に実験 1A で見られた長期の効果が，類似した別の課題においても品質の改善をもたらすかを複数の画像分類課題を組み合わせて検討した．

5.5.2 自己補正の短期的効果

実験 1A と実験 1B での自己補正のステージ要因と正答率の分析から，次の結果が得られた．まず，自己補正をタスクに適用することで，現実のクラウドソーシングワーカから得られるタスク結果の品質が改善されることがわかった．実験 1A では trusted 条件にて有意な正答率の改善が見られたことから，自己補正における参考回答の提示は，タスク結果の品質改善のための重要な要素であることが示唆された．さらに，実験 1B では correct 条件にて有意な正答率の改善が見られたことから，参考回答の品質は重要であると考えられるが，参考回答の品質と自己補正によるタスク結果の改善の関係は，これらの実験からは不明である．

実験 1B の random 条件においても，ステージ２の成績がステージ１を下回る傾向は見られなかったため，何らかの手法に基いて参考回答を提示できる場合には，参考回答を提示することが有効であると考えられる．ただし，参考回答の内容や提示

の方法は，ワーカがタスクに継続して取り組む際の動機づけを左右する要因になると考えられるため注意が必要である．

Shah らはワーカが自己補正により真面目に取り組むための報酬アルゴリズムが，今回は作業を終えたワーカに対して定額の報酬を支払った．それにもかかわらず，タスク結果の品質改善が見られたことから，自己補正は独自の報酬アルゴリズムを導入することが難しい状況 (例えばワーカに対して一定の報酬を支払うことにのみ対応しているサービスを用いる場合など) においても有効な手法であると言える．

5.5.3 自己補正の長期的効果

実験 1A の結果から，ワーカが自己補正に連続で取り組むことで，ワーカ自身の回答品質の改善につながることが示唆された．また，回答品質の改善はテスト時期の pre-mid 間よりも mid-post 間で大きくなることから，改善にはある程度のタスク数が必要であることが分かる．ただし，今回の実験からはワーカの学習に必要なタクス数は自明でなく，これは各ワーカの状態や扱う課題などの要因に左右されると考えられる．

さらに実験 1B の結果から，自己補正に連続で取り組んだとしても，全体の傾向としてワーカ自身の回答品質の改善に繋がらない例があることが示された．実験 1B では絵画の画像を提示してその作者を推定する課題を扱ったが，全体を通して平均正答率が低く，学習効果も見られなかった．実験 1B では実験 1A よりも多くの学習タスクを割り当てたが，扱う課題によっては学習を促すことが難しいことが分かった．同様の課題についてより多くの学習タスクを割り当てることで，学習効果が見られる可能性は否定できない．ただし，ワーカが継続してタスクにより組みやすくするための支援が必要であると考えられ，例えば継続してタスクに取り組むことに対する報酬を与えるなどが挙げられる．

実験 1A, 実験 1B を通して，全体の傾向にかかわらず，一部のワーカは pre から post にかけて正答率が改善することを確認することが出来た．すべてのワーカが高い学習意欲を持つとは考えにくいため，学習効果が見られたワーカに注目して手法の評価をしたり，彼らを早期に発見する技術が重要である．

5.5.4 学習の転移について

実験 2 では，自己補正タスクによる学習の転移が生じるかを明らかにする実験を検討した．しかし，実験結果から学習の転移は見られなかった．自己補正タスクにおけるタスク設計やタスク割り当て，扱う課題の工夫などにより学習の転移を促すことができれば，クラウドソーシングプラットフォームにおけるワーカの育成に貢献できる可能性がある．そのため，学習の転移を促しやすいデータセットの選択手

法や学習タスク数の決定方法を研究することは重要であるが，これらは今後の課題とする．

5.5.5　正答率が改善したワーカの分析

実験 1B での分析から，弱い傾向ではあるものの，自己補正における回答の変更頻度と post テストの正答率には関連があることが示唆された．この知見は，今後タスクに取り組む過程で能力が向上する可能性を持つワーカの早期発見や，自らの考えで作業に取り組んでいないワーカの発見などいくつかの応用が考えられる．ただし，回答の変更頻度と post テストの成績および成長度合いの分布は，実験ごとに大きく異なる可能性が高い．よって，この知見に基づく人間ワーカの能力改善のための仕組みを検討するためにはより多くの実験的な検証が必要である．

5.5.6　今後の課題

自己補正タスクの繰り返しについて　本研究の実験では，自己補正タスクを数十回連続で割り当てることにより，ワーカの正答率が改善されることを示した．ただし，学習に必要な繰り返し数については明らかではない．十分な学習に必要な繰り返し数は，ワーカや取り扱う課題などの要因により変化することが予想される．

画像分類課題以外への応用　本研究では，すべての実験において画像分類課題による実験を行った．一方で，画像分類課題以外の課題に対して自己補正を適用した場合にも同様の傾向が見られるかは注目すべき課題である．

インセンティブ設計との組み合わせ　本研究では，すべての実験において定額の報酬をワーカに対して支払った．かかわらず自己補正の短期的・長期的効果が観察されたことから，動的な報酬の変更が困難であるようなクラウドソーシングプラットフォームに置いても自己補正が有効であると考えられる．一方で，Shah らは自己補正を提案した論文にて，自己補正のための報酬設計アルゴリズムを提案している [62]．この報酬アルゴリズムを始めとする，既存の報酬アルゴリズムを自己補正に適用した場合の，短期的および長期的効果を検証することは興味深い課題の１つである．

報酬アルゴリズムを導入することにより，ワーカに対してより多くの学習タスクを割り当てたり，単に他者回答へと変更することを抑制することができると考えられる．

参考回答の選び方　実験 1A では，自己補正のステージ２で提示する参考回答として，課題の正答率に基づいて信頼性の高いワーカを選択肢，彼らの回答を用いた．実験

1B では，参考回答として正答やランダムな回答を用いた．しかし，現実の正解が未知である課題を扱うクラウドソーシングにおいて，信頼性の高い回答を得ることは困難であることが多い．クラウドソーシングではゴールドスタンダードクエスチョンや多数決の結果などに基づいて信頼性の高い回答を得る方法が広く採用されていることから，自己補正における参考回答においてもこれらの方法を応用することが可能であると考えられる．

松原らは [66]，自己補正の参考回答として機械学習に基づく分類器の推論結果を提示することを試みた．参考回答として提示するための回答として機械学習モデルなどから得た推論結果を用いることは，別のワーカから回答を得るよりも低いコストで実現できるため現実的な手段の 1 つであると考えられる．

特に，本研究の Human ＋ AI クラウドの状況においては，人間ワーカと AI ワーカが同一のタスクに同時に取り組む．その過程で得られる AI ワーカからのタスク結果を人間ワーカの能力改善のために活用することは，全体としての効率化を図る上で重要である．

5.6 まとめ

本研究では，現実のクラウドワーカにおける自己補正の効果を明らかにするために，現実のクラウドワーカが自己補正タスクに取り組む実験を検討した．そして，自己補正がもたらす効果について，タスクの正答率や反応時間などを分析した．

実験 1A では，鳥の画像分類タスクに自己補正を適用し，自己補正で提示する参考回答の有無を比較する実験を行った．実験 1B では，絵画の分類タスクに自己補正を適用し，自己補正で提示する参考回答の品質を比較する実験を行った．これらの実験結果から，自己補正によって現実のクラウドワーカがタスク結果を改善できることが示唆された．実験 1A では参考回答を提示する条件に，実験 1B では参考回答として正答を提示する条件において有意な改善が見られたことから，自己補正によるタスク結果の品質改善において参考回答は重要な要素であると考えられる．さらに，ワーカが自己補正を繰り返すことで，ワーカ自身の回答品質が改善される可能性が示唆された．この傾向は実験 1A では条件間に有意差が認められたのに対し，実験 1B では条件間の有意差が認められなかったことから，取り扱う課題などの要因に左右されやすいと考えられる．一方で，実験 1B のタスク結果について，自己補正タスクでの回答変更率を分析したところ，回答変更率が高すぎず低すぎないワーカの post テストの成績および成長度合いが高い傾向があることが示唆された．

実験 2 では，実験 1A および実験 1B で見られた自己補正によるワーカの品質改善が，学習課題と評価課題が異なる場合にも同様の傾向が見られるかを調べたところ，有意な傾向は認められなかった．

以上の実験結果から，自己補正がクラウドソーシングにおける品質改善に対して，タスク結果の改善とワーカの能力改善という２つの側面から貢献できる可能性を示した．

第6章　結論

本論文では，人間と計算機処理の協調による効率的な課題解決を実現するために，人間＋ AI クラウドにおけるクラウドソーシングの品質管理の問題に取り組んだ．

第 3 章では，人間＋ AI クラウドにおける自動的なタスク割り当てについて検討した．リクエスタが設定した要求精度を満たすように人間ワーカおよび AI ワーカに対してタスクを割り当てることを目的として，自動的なタスク割り当てを実現するために，人間＋ AI クラウドタスク割り当て問題（HACTAP）を定義した．リクエスタの要求精度を満たしながら，AI ワーカへのタスク割り当て数を最大化するために，AI ワーカから得られるタスク結果の全体を評価する代わりに，タスク結果のラベルが同じであるようなタスクの部分集合（タスククラスタ）毎に AI ワーカの統計的な評価を行う，Clusterwise Test-based Assignment (CTA) を提案した．さらに，CTA が各タスククラスタを独立に評価するのに対し，すでに人間ワーカおよび AI ワーカに割り当て済みのタスクおよび次の評価対象となるタスククラスタを考慮した全体的なタスク結果品質を評価する Global Test-based Assignment (GTA) を提案した．CTA および GTA により少なくとも要求精度を満たす割り当てを求められることについて理論解析を行った．ベンチマークデータセットおよび水害被害判定タスクを用いて，単一の AI ワーカの全体的な性能を評価するベースライン手法，および単一の AI ワーカを対象とした能動学習（ALA）を行うベースライン手法と提案手法の比較実験を行った．実験結果から，（1）CTA は ALA と比較して早い段階で AI ワーカへのタスク割り当てが可能であるが，最終的な AI ワーカへのタスク割り当て数は同程度であったこと（2）GTA は要求精度を下回るようなタスク結果品質をもたらす AI ワーカに CTA や ALA よりも多くのタスクを割り当てながら，全体としては要求精度を満たす割り当てが可能であることを示した．

第 4 章では，CTA において人間ワーカから得られるタスク結果品質が不正確な状況における振る舞いを分析した．人間ワーカのタスク結果は単なる成果物としてだけでなく，AI ワーカの学習と評価に利用される．したがって，人間ワーカから得られるタスク結果品質は特に重要であるが，現実のクラウドソーシングでは，様々な理由で人間ワーカからのタスク結果が不正確な可能性がある．これにより最終的なタスク結果品質の低下を引き起こす．そのため，多数決などの集約手法を適用することが一般的であるが，このコストをできるだけ削減したい．そこで，人間ワーカと AI ワーカの回答の不一致に着目し，人間ワーカの追加タスクの必要性を判定する

ことで，人間ワーカのタスク数の増加を抑えながらタスク結果品質を改善する手法を提案した．ベンチマークデータセットを用いた実験結果から，不確実な人間ワーカと AI ワーカが相互にタスク結果を共有することで，タスク結果品質を改善しながら効率的にタスクを処理する仕組みが構築可能であることを示した．

第 5 章では，Human ＋ AI クラウドにおける人間ワーカから得られるタスク結果の品質管理について議論した．第 4 章で議論したように，人間＋ AI クラウドタスク割り当ては，人間ワーカから得られるタスク結果を用いて AI ワーカの学習および評価を行うため，人間ワーカのタスク結果品質は Human ＋ AI クラウドから得られるタスク結果品質を左右する重要な要素である．多数決に代表されるタスク結果の集約手法を組み合わせることで，集約結果として得られるタスク結果品質を改善することは可能であるが，個々の人間ワーカから得られるタスク結果品質を向上させることが重要である．クラウドソーシングの分野において様々な研究がこの課題に取り組んでいるが，本研究ではその１つである Shah らが提案した自己補正に着目した．自己補正はタスク結果の品質を改善することを目的とした手法であり，１つのタスクに２度の回答の機会を与えることで，ワーカ自身が自分の誤答を補正できるのが特徴である．自己補正は，多数決をはじめとするタスク結果の集約手法や，優れたワーカの選出，タスク割り当て手法などと組み合わせることが容易であることから，多くの場面で活用できる可能性がある．しかし，自己補正が提案された論文では，タスク結果の品質改善についてのシミュレーションによる評価のみが行われており，現実のクラウドソーシングにおいてもシミュレーションと同様の効果が得られるかは不明であった．そこで，現実のクラウドワーカが自己補正を適用したタスクに取り組む実験により，自己補正がタスク結果やワーカにもたらす効果を検証した．実験結果は，自己補正が提案された論文のシミュレーション結果を支持するものであり，複数の実験から同様の傾向が見られた．さらに，ワーカが自己補正を繰り返すことにより，ワーカ自身の正答率が改善される長期的効果があることを実験から確認した．

6.1 本研究の貢献

本論文の貢献は次の通りである．

- 人間＋ AI クラウドの各ワーカへのタスク割り当ての問題（HACTAP）を定式化した．
- HACTAP に対して，依頼者の要求精度を満たすようなタスク割り当てを求めるアルゴリズムである Clusterwise Test-based Assignment (CTA) と Global Test-based Assignment (GTA) を提案し，アルゴリズムが決定したタスク割り当て

で得られるタスク結果品質が要求精度を満たすことを理論解析した．

- 人間ワーカから得られるタスク結果が不正確な条件では，CTA が要求精度を満たす割り当てを求められないこと，人間ワーカへのタスク割り当てに多数決を適用することでタスク結果品質を向上できることを実験的に示した．
- 人間ワーカと AI ワーカのタスク結果の不一致に基づいて，多数決による品質管理が必要なタスクを選択することで，タスク結果品質を低下させることなく，人間ワーカへのタスク割り当て数を削減できることを示した．このアイデアに基づく Interactive Clusterwise Test-based Assignment (ICTA) を提案した．
- 人間ワーカのタスク結果品質を改善するためのタスク設計手法である自己補正タスクについて，現実の人間ワーカによる実験を行い，その有効性を示した．
- 人間ワーカが自己補正タスクに繰り返し取り組むことで，人間ワーカのスキル向上に繋がることを実験的に示した．

本研究の成果は，大量のデータの処理に早急に取り組む必要があるが，処理の開始時点では適当な処理方法や計算機処理を行うべき処理工程が不明である場合に特に有効であると考えられる．そのような状況において，人間ワーカに作業を依頼するためのクラウドソーシングプラットフォームと機械学習モデルの集約ツールやコンペティションプラットフォームなどを本研究のタスク割り当てアルゴリズムに基づいて組み合わせることで，まずは人間ワーカによって処理を直ちに開始し，人間ワーカから得られたデータを用いて AI ワーカの構築および評価・割り当てを順次行うことが可能である．

既存の機械学習モデルコンペティションプラットフォームでは，参加者が提出するモデルの全体の性能が評価対象となり，性能が上位のモデルを作成した参加者に報酬が与えられる．本研究の枠組みにおいても全体的な正答率が高い AI ワーカの価値は高いが，それだけに限らない．なぜなら，全体的な性能が十分でなくとも，タスククラスタの品質が認められればその部分のタスク結果を採用することができるからである．このことから，不特定多数のプログラマーがリクエスタの課題に直接的に貢献できる機会が増えるのではないかと考えられる．さらに，単に正答率の高い AI ワーカを構築するのではなく，より多くのタスククラスタが採用されるような最適化を検討するといった方向性も想定される．これは例えば報酬設計の工夫などにより AI ワーカの開発者の行動を促すことが有効な手段の一つと言える．

6.2 今後の課題

本研究の今後の課題について述べる．

タスクの種類の拡張　本研究の実験では，画像を対象とする分類タスクを扱った．そのためテキストや音声，動画などの種類のデータを対象とする分類タスクでの有効性は未確認である．３章および４章で取り組んだタスク割り当てアルゴリズムは，分類タスクで扱う対象には依存していないため，これらの種類のデータにも適用可能であると考えられる．５章で扱った，自己補正に基づくタスク設計によりワーカの能力を向上させるという観点では，分類タスクで扱うデータの性質によって学習やタスクの回答の行動が変化する可能性がある．

さらに，本研究の成果をオブジェクト検知 [67] やセマンティックセグメンテーション [68] などの別のタスクへ応用することも重要な課題である．これらの課題は，対象となる画像の各ピクセルに注目すれば分類課題と見なすことができる．そこで，HACTAP のアルゴリズムにおいてタスククラスタの構成方法や評価方法に工夫を導入することで，これらの課題に応用できると考えられる．

金銭コストやタスクの処理速度の考慮　本研究では一貫してタスク結果の品質を手法の指標として，タスク割り当てアルゴリズムやタスク設計の有効性を評価した．しかし，現実のクラウドソーシング環境では，各ワーカがタスクを処理する際の具体的な金銭的コストや，タスクを割り当ててから結果を入手できるまでの時間が大きく異なるなど，より複雑な状況を考慮する必要がある．HACTAP における要求精度のように，コスト要件を制約として定めて，それに応じてタスク割り当てを最適化するようにアルゴリズムを改良することでこの問題に対処できると考えられる．ただし，特に AI ワーカに対する報酬設計には議論の余地があり，AI ワーカをどのように実行するか（AI ワーカ開発者が所有する計算機またはクラウドソーシングプラットフォーム）などにより大きく変化する可能性がある．

タスクを割り当ててから結果を入手するまでの時間を考慮して，できるだけ早く全てのタスクを完了したい場合，タスク処理の並列化を検討することが有望である．本論文で提案した CTA は各タスククラスタを独立に評価するため，並列可能性がベースライン手法の ALA や割り当て済みのタスククラスタと次の候補タスククラスタを考慮する GTA よりも並列可能性が高い．最適なタスク割り当てを求めることと，並列性を高めることの間にはトレードオフの関係があると考えられる．このトレードオフの問題にどう対処するかは重要な今後の課題である．

多様な AI ワーカの公募　本論文では，AI ワーカとして多様なアルゴリズム等に基づくワーカが参加することを想定し，それらを組み合わせて活用する方法を議論した．一方で，深層学習モデルなどの作成をクラウドワーカに依頼した場合に，提出される可能性が高いのは，機械学習ライブラリのサンプルコードなどに変更を加えた単純なモデル，あるいはインターネット上に公開されている学習済みモデルを fine-tuning

したモデルなどの少ない作業量で構築できるプログラムである．

報酬設計や AI ワーカ開発者への AI ワーカ開発の依頼方法の工夫により，多様な性質を持つ AI ワーカを入手するための方法を検討することは重要な課題である．

スパム AI ワーカへの対応　本論文では，悪意のある AI ワーカに対処する問題は扱わなかった．本論文が提案したアルゴリズムは，評価時の性能でタスクを処理すると仮定しているため，例えば評価対象のタスクと未ラベルのタスクに対して全く異なる方法でラベルを返す AI ワーカはアルゴリズムの意図しない挙動をしたことになる．

そのため，意図しない AI ワーカにタスクを割り当てたり意図しない報酬を支払わないために，AI ワーカの性能を継続的に評価したり，別の AI ワーカや人間ワーカのラベルとの比較，クラウドソーシングプラットフォーム側で性能評価の参考となる AI ワーカやタスククラスタを提供するなどの対策が考えられる．

謝辞

本論文を完成させるにあたり，多くの方々からのご指導とご支援をいただきました．

主研究指導担当教員である森嶋厚行先生には，私が学類生として森嶋研究室（融合知能デザイン研究室）に参加してから６年間に渡り，終始熱心なご指導を賜りました．森嶋先生は研究活動の素晴らしさだけでなく，相手に物事を正確に伝えるための文章の書き方やプレゼンの方法などの大切さについても丁寧にご指導くださり，またそれらを実践するための多くの機会を与えてくださいました．深く感謝申し上げます．

森田ひろみ先生には，博士前期課程および博士後期課程における副研究指導教員を引き受けていただきました．特に，研究活動を通して実験デザインや実験結果の分析手法について丁寧なご指導とご助言をいただきました．心より感謝申し上げます．

若林啓先生には，本論文の副研究指導教員を快く引き受けていただきました．特にタスク割り当てアルゴリズムの開発やその性能評価において丁寧なご指導とご助言をいただきました．心より感謝申し上げます．

鈴木伸崇先生，天笠俊之先生には，ご多忙にも関わらず本論文の審査委員を引き受けていただき，多くのご助言やご指摘をいただきました．心より感謝申し上げます．

融合知能デザイン研究室の松原正樹先生，伊藤寛祥先生，田中和世先生には，研究活動や論文執筆を進める上でのさまざまなご助言やご指導をいただきました．同研究室のメンバーおよびスタッフの皆様には研究生活全体を通して大変お世話になりました．

永森光晴先生，三原鉄也先生，阪口哲男先生，杉本重雄先生および各研究室の学生の皆様には，合同ゼミを通して特に研究発表について多くのご助言をいただきました．

Crowd4U の開発・運用を行う FusionComp のメンバーの皆様には大変お世話になりました．チームでソフトウェアを継続的に開発することの難しさや，研究成果を社会実装することの面白さなど，多くのことを学ばせていただきました．

本論文で取り組んだ各実験では，世界中のクラウドワーカの皆様にご協力をいただきました．彼らの協力により，この研究活動はより充実したものとなりました．ここに感謝いたします．また本研究において作成したソフトウェアやシステムは，さまざまな OSS プロジェクトを活用することで実現できました．各 OSS のコミュニティの皆様に感謝します．

本論文の研究の一部は国立研究開発法人 科学技術振興機構の AIP チャレンジプログラムによる支援を受けました．ここに感謝申し上げます.

最後に，大学院への進学を認めてくださり，そして研究生活を支えてくれた家族と友人の皆様に感謝申し上げます.

参考文献

[1] J HOWE. The rise of crowdsourcing. *Wired Magazine*, Vol. 14, No. 6, pp. 176–183, 2006.

[2] Luis Von Ahn, Benjamin Maurer, Colin McMillen, David Abraham, and Manuel Blum. recaptcha: Human-based character recognition via web security measures. *Science*, Vol. 321, No. 5895, pp. 1465–1468, 2008.

[3] Seth Cooper, Firas Khatib, Adrien Treuille, Janos Barbero, Jeehyung Lee, Michael Beenen, Andrew Leaver-Fay, David Baker, Zoran Popović, et al. Predicting protein structures with a multiplayer online game. *Nature*, Vol. 466, No. 7307, pp. 756–760, 2010.

[4] Edith Law and Luis von Ahn. Human computation. *Synthesis lectures on artificial intelligence and machine learning*, Vol. 5, No. 3, pp. 1–121, 2011.

[5] Klaas-Jan Stol and Brian Fitzgerald. Two's company, three's a crowd: A case study of crowdsourcing software development. In *Proceedings of the 36th International Conference on Software Engineering*, ICSE 2014, p. 187–198, New York, NY, USA, 2014. Association for Computing Machinery.

[6] Hisashi KASHIMA, Yukino BABA, Kashima Hisashi, and Baba Yukino. Human computation(<special issue>human computation and crowdsourcing). *Journal of the Japanese Society for Artificial Intelligence*, Vol. 29, No. 1, pp. 4–11, jan 2014.

[7] Florian Daniel, Pavel Kucherbaev, Cinzia Cappiello, Boualem Benatallah, and Mohammad Allahbakhsh. Quality control in crowdsourcing: A survey of quality attributes, assessment techniques, and assurance actions. *ACM Comput. Surv.*, Vol. 51, No. 1, pp. 7:1–7:40, January 2018.

[8] Srikanth Jagabathula, Lakshminarayanan Subramanian, and Ashwin Venkataraman. Reputation-based worker filtering in crowdsourcing. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger,

editors, *Advances in Neural Information Processing Systems 27*, pp. 2492–2500. Curran Associates, Inc., 2014.

[9] Lora Aroyo and Chris Welty. Measuring crowd truth for medical relation extraction. In *AAAI 2013 Fall Symposium on Semantics for Big Data*, 2013.

[10] Shayan Doroudi, Ece Kamar, Emma Brunskill, and Eric Horvitz. Toward a learning science for complex crowdsourcing tasks. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pp. 2623–2634. ACM, 2016.

[11] Peter Kinnaird, Laura Dabbish, Sara Kiesler, and Haakon Faste. Co-worker transparency in a microtask marketplace. In *Proceedings of the 2013 conference on Computer supported cooperative work*, pp. 1285–1290. ACM, 2013.

[12] Gary Hsieh and Rafał Kocielnik. You get who you pay for: The impact of incentives on participation bias. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*, pp. 823–835. ACM, 2016.

[13] Alexander Philip Dawid and Allan M Skene. Maximum likelihood estimation of observer error-rates using the em algorithm. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, Vol. 28, No. 1, pp. 20–28, 1979.

[14] Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. Learning with noisy labels. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, Vol. 26. Curran Associates, Inc., 2013.

[15] 吉屹李, 雪乃馬場, 久嗣鹿島. 超問題：専門知識を要するクラウドソーシングタスクの回答統合法. 日本データベース学会和文論文誌, Vol. 17-J, No. 8, 3 2019.

[16] Nguyen Quoc Viet Hung, Nguyen Thanh Tam, Ngoc Tran Lam, and Karl Aberer. An evaluation of aggregation techniques in crowdsourcing. In *Web Information Systems Engineering - WISE 2013 - 14th International Conference, Nanjing, China, October 13-15, 2013, Proceedings, Part II*, pp. 1–15, 2013.

[17] Burr Settles. Active learning literature survey. Technical report, University of Wisconsin-Madison Department of Computer Sciences, 2009.

[18] Yan Yan, Romer Rosales, Glenn Fung, and Jennifer G. Dy. Active learning from crowds. In *Proceedings of the 28th International Conference on Interna-*

tional Conference on Machine Learning, ICML'11, pp. 1161–1168, USA, 2011. Omnipress.

[19] Natsumaro Kutsuna, Takumi Higaki, Sachihiro Matsunaga, Tomoshi Otsuki, Masayuki Yamaguchi, Hirofumi Fujii, and Seiichiro Hasezawa. Active learning framework with iterative clustering for bioimage classification. *Nature communications*, Vol. 3, p. 1032, 2012.

[20] Jie Yang, Thomas Drake, Andreas Damianou, and Yoelle Maarek. Leveraging crowdsourcing data for deep active learning an application: Learning intents in alexa. In *Proceedings of the 2018 World Wide Web Conference*, WWW '18, pp. 23–32, Republic and Canton of Geneva, Switzerland, 2018. International World Wide Web Conferences Steering Committee.

[21] Patrick Jörger, Yukino Baba, and Hisashi Kashima. Learning to enumerate. In Alessandro E.P. Villa, Paolo Masulli, and Antonio Javier Pons Rivero, editors, *Artificial Neural Networks and Machine Learning – ICANN 2016*, pp. 453–460, Cham, 2016. Springer International Publishing.

[22] Thomas G. Dietterich. Ensemble methods in machine learning. In *Multiple Classifier Systems*, pp. 1–15, Berlin, Heidelberg, 2000. Springer Berlin Heidelberg.

[23] W. H. Beluch, T. Genewein, A. Nurnberger, and J. M. Kohler. The power of ensembles for active learning in image classification. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9368–9377, Los Alamitos, CA, USA, jun 2018. IEEE Computer Society.

[24] Ece Kamar, Severin Hacker, and Eric Horvitz. Combining human and machine intelligence in large-scale crowdsourcing. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems - Volume 1*, AAMAS '12, p. 467–474, Richland, SC, 2012. International Foundation for Autonomous Agents and Multiagent Systems.

[25] O. Russakovsky, L. Li, and L. Fei-Fei. Best of both worlds: Human-machine collaboration for object annotation. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2121–2131, June 2015.

[26] Akshay L Chandra, Sai Vikas Desai, Vineeth N Balasubramanian, Seishi Ninomiya, and Wei Guo. Active learning with point supervision for cost-effective panicle detection in cereal crops. *Plant Methods*, Vol. 16, No. 1, pp. 1–16, 2020.

[27] Ines Arous, Jie Yang, Mourad Khayati, and Philippe Cudre-Mauroux. Peer grading the peer reviews: A dual-role approach for lightening the scholarly paper review process. In *Proceedings of the Web Conference 2021*, WWW '21, p. 1916–1927, New York, NY, USA, 2021. Association for Computing Machinery.

[28] Yolanda Gil, James Honaker, Shikhar Gupta, Yibo Ma, Vito D'Orazio, Daniel Garijo, Shruti Gadewar, Qifan Yang, and Neda Jahanshad. Towards human-guided machine learning. In *Proceedings of the 24th International Conference on Intelligent User Interfaces*, IUI '19, pp. 614–624, New York, NY, USA, 2019. ACM.

[29] Jinfeng Yi, Rong Jin, Shaili Jain, Tianbao Yang, and Anil Jain. Semi-crowdsourced clustering: Generalizing crowd labeling by robust distance metric learning. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, Vol. 25, pp. 1772–1780. Curran Associates, Inc., 2012.

[30] An Thanh Nguyen, Byron C Wallace, and Matthew Lease. Combining crowd and expert labels using decision theoretic active learning. In *Proceedings of the 3rd AAAI Conference on Human Computation and Crowdsourcing*. aaai.org, September 2015.

[31] Jie Yang, Alisa Smirnova, Dingqi Yang, Gianluca Demartini, Yuan Lu, and Philippe Cudre-Mauroux. Scalpel-cd: Leveraging crowdsourcing and deep probabilistic modeling for debugging noisy training data. In *The World Wide Web Conference*, WWW '19, pp. 2158–2168, New York, NY, USA, 2019. ACM.

[32] Bryan Wilder, Eric Horvitz, and Ece Kamar. Learning to complement humans. In Christian Bessiere, editor, *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, pp. 1526–1533. ijcai.org, 2020.

[33] Anthony Liu, Santiago Guerra, Isaac Fung, Gabriel Matute, Ece Kamar, and Walter Lasecki. Towards hybrid human-ai workflows for unknown unknown detection. In *Proceedings of The Web Conference 2020*, WWW '20, p. 2432–2442, New York, NY, USA, 2020. Association for Computing Machinery.

[34] Radu Herbei and Marten H. Wegkamp. Classification with reject option. *The Canadian Journal of Statistics / La Revue Canadienne de Statistique*, Vol. 34, No. 4, pp. 709–721, 2006.

[35] 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.

[36] Donald Honeycutt, Mahsan Nourani, and Eric Ragan. Soliciting human-in-the-loop user feedback for interactive machine learning reduces user trust and impressions of model accuracy. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, Vol. 8, pp. 63–72, 2020.

[37] Zizhe Wang and Hailong Sun. Teaching active human learners. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35, pp. 5850–5857, 2021.

[38] 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.

[39] 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.

[40] 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.

[41] 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.

[42] 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.

[43] 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.

[44] 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.

[45] 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.

[46] 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.

[47] 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.

[48] 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.

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

[50] 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.

[51] 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.

[52] Jingbo Zhu, Huizhen Wang, Eduard Hovy, and Matthew Ma. Confidence-based stopping criteria for active learning for data annotation. *ACM Trans. Speech Lang. Process.*, Vol. 6, No. 3, April 2010.

[53] Adel Javanmard, Andrea Montanari, et al. Online rules for control of false discovery rate and false discovery exceedance. *The Annals of statistics*, Vol. 46, No. 2, pp. 526–554, 2018.

[54] Tarin Clanuwat, Mikel Bober-Irizar, Asanobu Kitamoto, Alex Lamb, Kazuaki Yamamoto, and David Ha. Deep learning for classical japanese literature. *CoRR*, Vol. abs/1812.01718, , 2018.

[55] Masaki Kobayashi, Kei Wakabayashi, and Atsuyuki Morishima. Quality-aware dynamic task assignment in human+ai crowd. In *Companion of The 2020 Web Conference 2020*, pp. 118–119, 2020.

[56] Masaki Kobayashi, Kei Wakabayashi, and Atsuyuki Morishima. Human+ai crowd task assignment considering result quality requirements. *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, Vol. 9, No. 1, 2021.

[57] E.F. Moore and C.E. Shannon. Reliable circuits using less reliable relays. *Journal of the Franklin Institute*, Vol. 262, No. 3, pp. 191 – 208, 1956.

[58] Tarin Clanuwat, Mikel Bober-Irizar, Asanobu Kitamoto, Alex Lamb, Kazuaki Yamamoto, and David Ha. Deep learning for classical japanese literature. *CoRR*, Vol. abs/1812.01718, , 2018.

[59] Eleanor Jack Gibson. Principles of perceptual learning and development. 1969.

[60] James J Gibson and Eleanor J Gibson. Perceptual learning: Differentiation or enrichment? *Psychological review*, Vol. 62, No. 1, p. 32, 1955.

[61] Everett Mettler and Philip J Kellman. Adaptive response-time-based category sequencing in perceptual learning. *Vision research*, Vol. 99, pp. 111–123, 2014.

[62] 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.

[63] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona. Caltech-UCSD Birds 200. Technical Report CNS-TR-2010-001, California Institute of Technology, 2010.

[64] David N Perkins, Gavriel Salomon, et al. Transfer of learning. *International encyclopedia of education*, Vol. 2, pp. 6452–6457, 1992.

[65] Susan M Barnett and Stephen J Ceci. When and where do we apply what we learn?: A taxonomy for far transfer. *Psychological bulletin*, Vol. 128, No. 4, p. 612, 2002.

[66] Masaki Matsubara, Masaki Kobayashi, and Atsuyuki Morishima. A learning effect by presenting machine prediction as a reference answer in self-correction. *In Proceedings of The Second IEEE Workshop on Human-in-the-loop Methods and Human Machine Collaboration in BigData (IEEE HMData2018)*, pp. 3521–3527, 2018.

[67] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pp. 740–755, Cham, 2014. Springer International Publishing.

[68] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 633–641, 2017.

研究業績の一覧

主著論文

論文誌

1-1. 小林 正樹, 若林 啓, 森嶋 厚行. 人間+AI Crowd の相互作用によるタスク結果品質の管理手法. 日本データベース学会和文論文誌 (20:2). 2022, 8 pages.

1-2. Masaki Kobayashi, Hiromi Morita, Masaki Matsubara, Nobuyuki Shimizu, and Atsuyuki Morishima. Empirical Study on Effects of Self-Correction in Crowdsourced Image Classification Tasks. Human Computation Journal (8:1). 2021, p. 1-24.

国際会議

口頭発表

2-1. Masaki Kobayashi, Kei Wakabayashi and Atsuyuki Morishima. Human+AI Crowd Task Assignment Considering Result Quality Requirements. The 9th AAAI Conference on Human Computation and Crowdsourcing (HCOMP 2021). Virtual Conference, 2021, 11 pages.

2-2. 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. Proceedings of the Sixth AAAI Conference on Human Computation and Crowdsourcing (HCOMP 2018). Zurich, Switzerland, 2018, p. 79-87.

ポスター発表

3-1. Masaki Kobayashi, Kei Wakabayashi, and Atsuyuki Morishima. Quality-Aware Dynamic Task Assignment in Human+AI Crowd. Companion Proceedings of the Web Conference 2020 (WWW '20). Taipei, Taiwan, 2020, p. 118–119.

国内会議・研究会

4-1. 小林 正樹, 若林 啓, 森嶋 厚行. 人間+AI クラウドの相互作用によるタスク結果品質の管理手法. 第 13 回データ工学と情報マネジメントに関するフォーラム (DEIM2021). Virtual Conference, 2021, 8 pages.

4-2. 小林 正樹, 若林 啓, 森嶋 厚行. タスク結果品質を考慮した人間+AI クラウドへのマイクロタスク割り当て. 第 12 回データ工学と情報マネジメントに関するフォーラム (DEIM2020). Virtual Conference, 2020, 8 pages.

4-3. 小林 正樹, 若林 啓, 森嶋 厚行. 人間+AI クラウドにおけるマイクロタスク処理の効率化. 第 12 回 Web とデータベースに関するフォーラム (WebDB Forum 2019). 東京都新宿区, 2019, p. 5-8.

4-4. 小林 正樹, 森田 ひろみ, 松原 正樹, 清水 伸幸, 森嶋 厚行. 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. 第 11 回データ工学と情報マネジメントに関するフォーラム (DEIM2019). 長崎県, 2019, 7 pages.

4-5. 小林 正樹, 森田 ひろみ, 松原 正樹, 清水 伸幸, 森嶋 厚行. クラウドワーカの品質改善における他者回答提示の短期的・長期的効果. 第 10 回データ工学と情報マネジメントに関するフォーラム (DEIM2018). 福井県, 2018, 8 pages.

4-6. 小林 正樹, 清水 伸幸, 森嶋 厚行. ワーカの成長を考慮した自己補正マイクロタスク割当て手法. 科学技術振興機構 CREST 3 プロジェクト合同シンポジウム (ポスター発表). 茨城県つくば市, 2017, .

4-7. 小林 正樹, 清水 伸幸, 森嶋 厚行. ワーカの成長を考慮した自己補正マイクロタスク割当て手法. 第 9 回データ工学と情報マネジメントに関するフォーラム (DEIM2017). 岐阜県, 2017, 6 pages.

4-8. 小林 正樹, 伏見 卓恭, 佐藤 哲司. 購買履歴を用いたユーザ行動モデルの推定. 第 8 回データ工学と情報マネジメントに関するフォーラム (DEIM2016). 福岡, 2016, 5 pages.

4-9. 小林 正樹, 伏見 卓恭, 佐藤 哲司. 調理手順の頻出パターンに基づく入力支援手法の提案. 信学技報 (データ工学研究会, データ工学と食メディア) (115:230). 2015, p. 53-57.

主著以外の論文

5-1. Munenari Inoguchi and Keiko Tamura and Kousuke Uo and Masaki Kobayashi and Atsuyuki Morishima. Time-Cost Estimation for Early Disaster Damage

Assessment Methods, Depending on Affected Area. Journal of Disaster Research (16:4). 2021, p. 733-746.

5-2. Munenari Inoguchi, Keiko Tamura, Kousuke Uo, and Masaki Kobayashi. Validation of CyborgCrowd Implementation Possibility for Situation Awareness in Urgent Disaster Response -Case Study of International Disaster Response in 2019-. 2020 IEEE International Conference on Big Data (IEEE HMData 2020). Virtual Conference, 2020, p. 3062-3071.

5-3. Akiko Aizawa, Frederic Bergeron, Junjie Chen, Fei Cheng, Katsuhiko Hayashi, Kentaro Inui, Hiroyoshi Ito, Daisuke Kawahara, Masaru Kitsuregawa, Hirokazu Kiyomaru, Masaki Kobayashi, Takashi Kodama, Sadao Kurohashi, Qianying Liu, Masaki Matsubara, Yusuke Miyao, Atsuyuki Morishima, Yugo Murawaki, Kazumasa Omura, Haiyue Song, Eiichiro Sumita, Shinji Suzuki, Ribeka Tanaka, Yu Tanaka, Masashi Toyoda, Nobuhiro Ueda, Honai Ueoka, Masao Utiyama and Ying Zhong. A System for Worldwide COVID-19 Information Aggregation. Proceedings of the 1st Workshop on NLP for COVID-19 (Part 2) at EMNLP 2020. Virtual Conference, 2020, 9 pages.

5-4. Yu Yamashita, Masaki Kobayashi, Kei Wakabayashi, and Atsuyuki Morishima. Dynamic Worker-Task Assignment for High-Quality Task Results with ML Workers. The eighth AAAI Conference on Human Computation and Crowdsourcing (HCOMP2020). Virtual Conference, 2020, 3 pages.

5-5. 山下 裕, 小林 正樹, 若林 啓, 森嶋 厚行. クラウドソーシングにおける AI を利用したタスク削減手法. 第 12 回データ工学と情報マネジメントに関するフォーラム (DEIM2020). Virtual Conference, 2020, 7 pages.

5-6. 鵜尾 厚佑, 小林 正樹, 松原 正樹, 馬場 雪乃, 森嶋 厚行. 階層型のラベル付けマイクロタスクにおける能動学習戦略の比較. 第 12 回データ工学と情報マネジメントに関するフォーラム (DEIM2020). Virtual Conference, 2020, 6 pages.

5-7. Kousuke Uo, Masaki Kobayashi, Masaki Matsubara, Yukino Baba, and Atsuyuki Morishima:. Active Learning Strategies for Hierarchical Labeling Microtasks. The 3rd IEEE Workshop on Human-in-the-loop Methods and Human Machine Collaboration in BigData (IEEE HMData 2019). Los Angeles, 2019, p. 4647-4650.

5-8. Masafumi Hayashi, Masaki Kobayashi, Masaki Matsubara, Toshiyuki Amagasa, and Atsuyuki Morishima. Incentive Design for Crowdsourced Development of

Selective AI for Human and Machine Data Processing: A Case Study. The 3rd IEEE Workshop on Human-in-the-loop Methods and Human Machine Collaboration in BigData (IEEE HMData 2019). Los Angeles, 2019, p. 4596-4601.

5-9. 松原 正樹, 小林 正樹, 森嶋 厚行. 機械学習の分類予測に基づく参考回答提示によるクラウドワーカの学習効果. 第 11 回データ工学と情報マネジメントに関するフォーラム (DEIM2019). 長崎県, 2019, 7 pages.

5-10. Masaki Matsubara, Masaki Kobayashi, Atsuyuki Morishima. A Learning Effect by Presenting Machine Prediction as a Reference Answer in Self-correction. The Second IEEE Workshop on Human-in-the-loop Methods and Human Machine Collaboration in BigData (IEEE HMData2018). Seattle, 2018, p. 3522-3528.

付録

３章の実験で用いた AI ワーカの一覧

３章のオープンベンチマークデータセットでの実験で利用した AI ワーカの一覧を次に示す．各項目は Python の機械学習ライブラリである scikit-learn に実装されている機械学習モデルのクラス名である．

1. MLPClassifier
2. ExtraTreeClassifier
3. LogisticRegression
4. KMeans (Used only by CTA/GTA)
5. DecisionTreeClassifier
6. SVC (probability=True option was used in WTA and ALA)
7. KNeighborsClassifier
8. GaussianProcessClassifier
9. MultinomialNB
10. AdaBoostClassifier
11. PassiveAggressiveClassifier (Used only by CTA/GTA)
12. RidgeClassifier (Used only by CTA/GTA)
13. RidgeClassifierCV (Used only by CTA/GTA)
14. ComplementNB
15. NearestCentroid (Used only by CTA/GTA)